

Cited Reference 1

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-101986

(43)Date of publication of application : 15.04.1997

(51)Int.Cl.

G06F 17/60
G06F 9/445

(21)Application number : 07-258507

(71)Applicant : FUJITSU LTD

(22)Date of filing : 05.10.1995

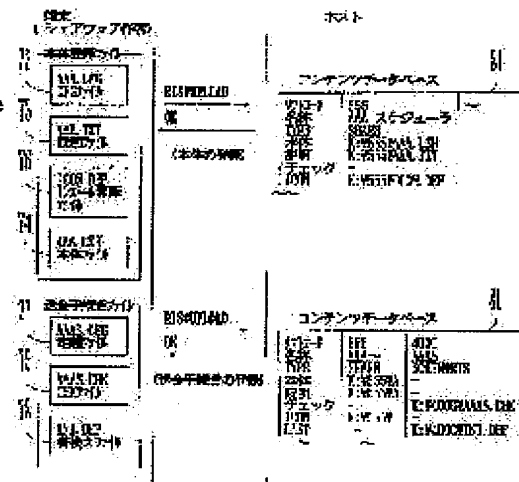
(72)Inventor : OKI HIROSHI
KAMATA SHINJI
HARA TAKASHI
OKADA TOSHIO
YAMAZAKI TOSHIYA

(54) SYSTEM AND METHOD FOR SETTLING ACCOUNT OF SOFTWARE

(57)Abstract:

PROBLEM TO BE SOLVED: To automatically clear the bill of a delivered software in the service of vending the software through a communication line.

SOLUTION: The designer of software up loads a main body registration file and a money transfer procedure file through a communication network to a host computer. The host computer registers this information into a contents data base 81 and when a delivery request is received from a user, charging/ delivery is performed corresponding to the type of software. In the case of shareware, for example, the drop of price from account and the cancel of function limitation are simultaneously performed in place of the information of money transfer procedure file. Besides, the host computer can variously set prices by storing a price file describing the algorithm of price decision.



LEGAL STATUS

[Date of request for examination] 11.07.2002

[Date of sending the examiner's decision of rejection] 24.05.2005

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3711162

[Date of registration] 19.08.2005

[Number of appeal against examiner's decision of rejection] 2005-11641

[Date of requesting appeal against examiner's decision of rejection] 21.06.2005

[Date of extinction of right]

* NOTICES *

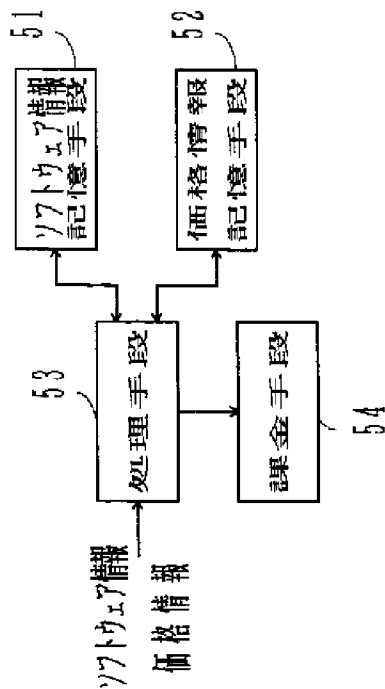
JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.**** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DRAWINGS

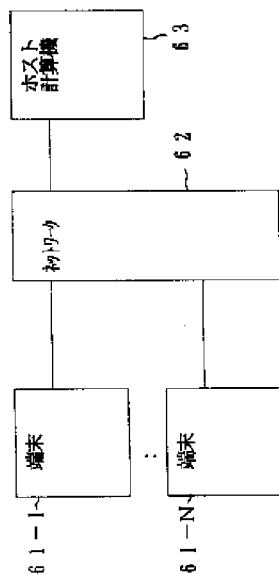
[Drawing 1]

本発明の原理図



[Drawing 2]

実施形態のシステム構成図



[Drawing 7]

説明ファイルを示す図

新ワープロ-A2は、当社が最新の技術をもって開発した、中国語まで対応出来るワープロです....

[Drawing 8]

本体ファイルを示す図

NEWWP.LZH=WP.EXE+NEWWP.ICO+README.DOC
 NEWWPLIB.LZH=GSWDDL.DLL

[Drawing 9]

第1のCHKファイルを示す図

```

MQ1 FILE "WP.EXE";
MA1 "NONE" then DLOAD
    else IC#SOUKO="@ST4DIR0";
MQ2 FILE "WP.EXE";
MA2 @ST4DATE0 < "1995/06/30 00:00:00" then SQ2;
    else SQ1;
SQ1 QUERY "貴方のマシンに、既に、最新版が入ってます";
SA1 "強行" then RIS;
    "不要" then NG;
SQ2 FILE "GSWDL.DLL";
SA2 "NONE" then active=S2, active=C1, RIS;
    else RIS;

```

[Drawing 13]

第2のCHKファイルを示す図

CHKファイル—AAAS.CHK

```

MQ1 FILE "a" "AAA.INI";
MA1 "EXIST": SQ1;
    "NONE": SQ2;
SQ1 GETFIL "あなたのAAA.INIの場所を入力してください。" "ST4DIR0";
SA1 CANCEL: NG;
SA1 DEFAULT: IC#TARGET=0RET0&SOKIN#RIS;
SQ2 QUERY "あなたのシステムにはAAA.INIがありません。";
SA2 "はい": NG;

```

[Drawing 3]

CFGファイルを示す図 (その1)

```
[name]
;ソフト検索用名称          — 検索DBに登録
;フォーマットは,"名称","読み"とする。名称、読みとも、'"'で括る。
;新ワープロ-A2","シンワープロエーニ"

[version]
;ASCII 8文字以内          — ソフトDBに登録
V1.2

[oldversion]
;旧バージョンソフトの登録
;フォーマットは"ソフトコード バージョン文字列"
V1.1

[type]
;製品は、ダウンロード開始時に料金徴収される
;FREE
;フリーウェア
SHARE
;シェアウェア
;PRODUCT
;製品

[price]
;価格を数字のみで記す。頭に'¥'をつけたり、数字を','で区切ったりしてはいけない
3000

[machine]
;フォーマットはマシン識別子 マシン名          — インストールDBに登録
;ホストが必要なのはマシン識別子。マシン名は登録者のために添付
;動作可能マシンのみコメントをはずす
TOWNS
; Townsシリーズ
FMR
; FMRシリーズ
PC98
; PC98シリーズ
PC-AT
; AT互換機
;MAC1
; Quadra 800
;MAC2
; LC50
```

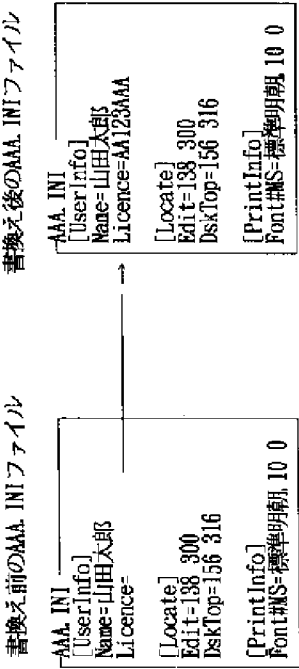
[Drawing 10]

ICONファイルを示す図

```
[ICON]
GROUP=RIS倉庫
TITLE=新ワープロ-A2
EXE=@IC#SOUK00*WP. EXE
CWD=@IC#SOUK00
WICON=@IC#SOUK00*NEWWP. ICO
TEXTTYPE=WIN
```

[Drawing 20]

ファイルの書換えを示す図



[Drawing 4]

CFGファイルを示す図 (その2)

```

[os]
;[machine]と同様      —インストールDBに登録
WIN31J
; 日本語 Windows 3.1
;NT31J
; 日本語 Windows NT 3.1
;DOS5V
; DOS/V Ver. 5.x
;DOS5J
; 日本語MS-DOS Ver. 5.0
;DOS3J
; 日本語MS-DOS Ver. 3.x
;TOS
;TOWNS OS
;MACOS7
; MacOS Ver. 7.0
[key]

;フォーマットは、"キー1","キー2",...,"キーn"を
;一行に書く。複数指定可能      —検索DBに登録
;"中国語"
[compression]
;圧縮の形式を示す。コメントの付け変えのみ可
;NONE
; 圧縮なし
LHA
; LHA による圧縮
;TGZ
; gnu tar + gzipによる圧縮
;ZIP
; PKZIP による圧縮
[instype]
;インストールタイプ、許さないTYPEをコメントアウト
;フォーマットは、"ID [iconfile1 [iconfile2 [...]]]"とする。
RIS      icon.def
; install してシステムに登録
INSTALLER icon2.def
; INSTALL.EXE自体をICON登録
DLOAD
; ダウンロード

```

[Drawing 14]

書換えファイルを示す図

書換えファイルINI. DEF
 <@IC#TARGET@YAAA. INI> ;ユーザシステムのAAA. INIファイルを書き換える。
 [User Info] ;[User Info]セクションのLicenceの値をAA123AAAにする
 Licence=AA123AAA

[Drawing 34]

リモートインストールプロトコルを示す図(その2)

```

RIS_CHKENV*RESP(ホスト→ 端末)
RID=6,
JUDGE=1,
STRPLACE:
  SOFT=5,
  WORKDIR="D:\RIS\SOUKO",
  SOUKODIR="D:\RIS\SOUKO\FM"  <-----サブディレクトリ付
;
WORKSIZ=309000,
SOUKOSIZ=549

```

[Drawing 5]

C F Gファイルを示す図 (その3)

[souko]

;ソフトを格納する推奨ディレクトリを記述する。@SOUKO@で始まることが望ましい。
;フォーマットは、"ID ディレクトリ名"で、"ID"は、"P"で始まり数字が続く("Pn")。

P1 @IC#SOUKO@ABC

P2 @WORK@

[source]

;フォーマットは、"ID FILE ExtPath mode arsize extsize extfiles"とする。

;ID: S1... Sn の nは連番を記述

;FILE:ダウンロードする本体ファイル(LZH形式のはず)名を記述

;ExtPath:展開すべきディレクトリ名を記述。特殊ディレクトリ記述あり

;ファイル名はASCIIのみ許す

;mode:端末側で、展開先ディレクトリを変更していいかどうかを示す。

; M:変更可能(Movable) F:変更不可能(Fixed)

;arsize:アーカイブの容量(bytes)

;extsize:展開後の最低必要量(bytes)

;インストール時に大量にWork領域を必要とする場合は、arcのサバを読む

;実行時に大量にWork領域を必要とする場合は、minのサバを読む

;ただし、他のソフトのインストールにより、領域不足となる可能性あり

;extfiles: FILEで示されるアーカイブファイルの中に圧縮格納されているファイル名
を、で区切って記述する。全てのファイルを書く必要はなく、端末で重
複チェックを行ないたいものだけ記述すればよい。従って、省略は可能と
する。なお、途中で改行を入れてはいけない。必ず全てのファイルを一行
に記述すること。

;ID FILE ExtPath mode arsize extsize extfiles

S1 NEWWP.LZH @P1@ M 800000 2000000

*S2 NEWWPLIB.LZH @P2@ F 7000 13000 GSWDLL.DLL

[destination]

;フォーマットは、"ID ディレクトリ名"で、"ID"は、"D"で始まり数字が続く("Dn")。
;ファイルを展開する時の展開先ディレクトリを列挙。基本的に特殊ディレクトリのみ
記述可 —インストールDBに登録

D1 @WINSYSTEM@

[copy]

;移動させるべきファイルや、Mode変更(場所移動なしでも)がある場合のみ記述

;[destination]セクションと同時に使用してもよいが、直接移動先ファイル名(特殊
ディレクトリを含むフルパス)を書いてもよい。

;フォーマットは、"ID @SID@SourcePath @DID@DestPath Attr"

;モードを変更するだけで、移動がない場合のDestinationは、'*'とする。

;ID:順に C1, C2, ..., Cn とする。チェックスクリプトで参照可能

;SID: [source]で記述した ID

;SourcePath:展開後現れるファイル名(サブディレクトリを含んでよい)

;DID: [destination]で記述した ID

;DestPath:展開後現れたファイルの移動先ファイル名(サブディレクトリを含んでよい)

;Attr:移動後隠す/書き込み禁止などを実行("attrib +n"のみ実行。"-n"はなし)

; R:RO H:Hidden A:Archive S:System N:Noneを続けて記述

;何も設定しない時は、省略可

;ID Source Destination Attr

*C1 @S2@GSW DLL @D1@GSW DLL N

[Drawing 6]

CFGファイルを示す図 (その4)

[first]

インストールの際に、アーカイブファイルのダウンロード前に実行しておきたいコマンドを記述する。例えば、ファイルのリネームなどである。

形式は、"ID TYPE COMMAND ARGS FILES"とする。

ID: "Fn" (n = 1, 2, 3, ...) と記述する。

頭に '*' をつけた場合は、"Sn", "Cn" と同じでデフォルトでは実行しません。

チェックスクリプトで ACTIVATE=Fn とした場合のみ実行します。

TYPE: S, C, D, L のいずれかを記述。それぞれ、S[cript], C[ommand], D[ownload], L[ha] の略である。

Script: インストールスクリプトに、サブルーチン的に機能を付け加えておき、それを呼び出す。ホスト側で機能を想定して作成しておく。現在は未対応。

Command: 端末プログラム配布時にある程度想定したコマンドを送りつけておき(ST4, [m]#reso などと同じ)、それを実行する。もしくは、他のデータなどと一緒に送った LZH ファイルの中に予め仕込んであったものを実行する。

Download: 個別にプログラムをダウンロードし、実行

Lha: Download の一種だが、ホスト上では lha で固めてある。FILES 欄には、*.LZH を一つだけ書ける(この固まりにプログラム、データをまとめて入れておく)。

つまり、この TYPE は、コマンドを実行する際に、そのコマンドが「どこ」にあるかを指定するものである。当面、"C", "D" の利用を想定している。

COMMAND: 実行すべきコマンドを記述する。注意点として、

1) TYPE が S, D, L の場合は、コマンド名だけで良い。

この場合の LZH は、ワークディレクトリに展開されるものとする。LZH で端末にファイルが渡された場合、後始末で *.lzh 以外のファイルを消せない(ホストは lha で何が展開されるかわからない)ので、"Fn" を使って自分で削除定義をする必要がある。

2) TYPE が C の場合には、DLOAD による *.LZH に仕込んであった場合は、その展開場所を変数で指定。端末プログラムとして配布済みの場合は、コマンド名だけで良い(パスに記述されてるから)。

ARGS: COMMAND に与える引数を、" " で括って記述する。変数(0...0)を使用してよい。引数がない場合は、空文字を"" で括っておく。

FILES: 必要なファイルが1つとは限らないので、' ' で区切って複数記述して良いこととする。全てのファイルのダウンロードが終了した後に、COMMAND を実行する。

ID	TYPE	COMMAND	ARGS	FILES
F1	C	BACKRIS	"WP.EXE"	BACKRIS.EXE

[last]

インストールの際に、アイコン登録コマンド実行後に実行したいコマンドを記述する。

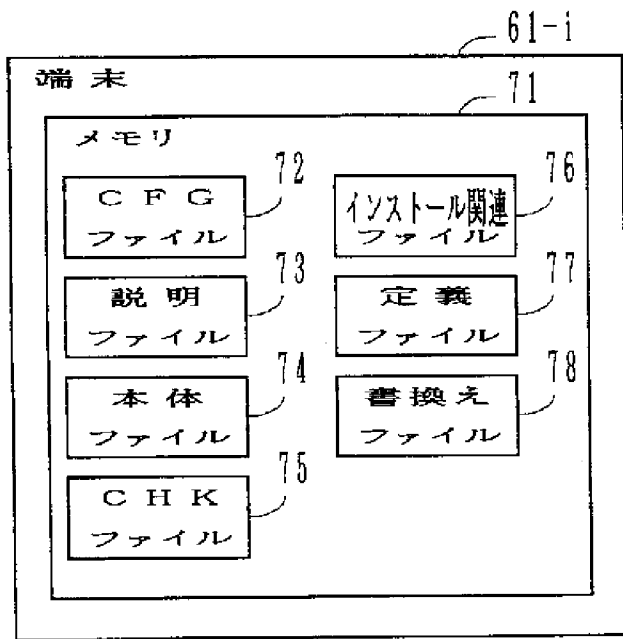
例えば、AUTOEXEC.BAT, CONFIG.SYS, WIN.INI の書き換えなどである。

形式は、[first] と同様。ただし、ID は、"Ln" (n = 1, 2, 3, ...) と記述する。

ID	TYPE	COMMAND	ARGS	FILES
L1	D	CHANGE	"AUTO.REG @BOOT0:%autoexec.bat @S10%autoexec.bat"	

[Drawing 11]

端末上で作成されるファイル群を示す図



[Drawing 12]

第 1 の定義ファイルを示す図

定義ファイル—AAAS.CFG—

```
[name]
"AAAスケジューラ","エースリー"

[type]
SOKIN 555 FJOKI AAA ;ソフトウェア555に対する送金手続き
                        送金はID:FJOKIに行う

[price]
4000

[machine]
TOWNS
FMR
PC98
PC-AT

[os]
WIN31J

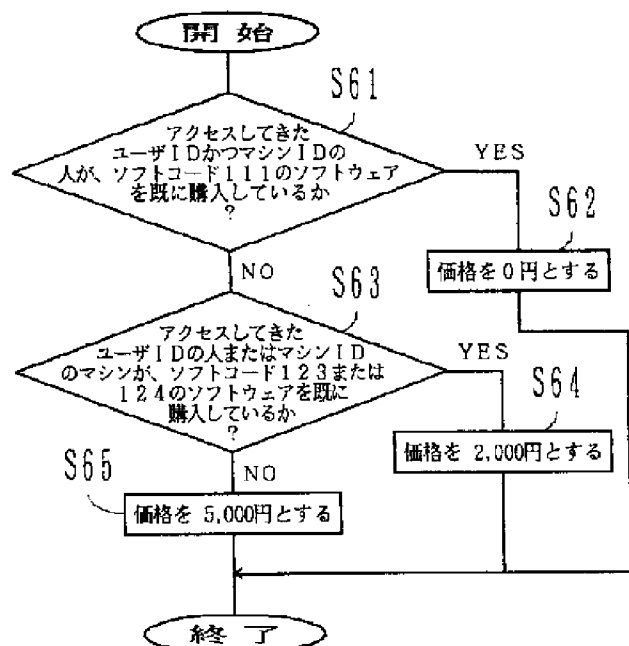
[key]
"T実用ソフト","Tスケジューラ"

[instype]
SOKIN#RIS ;送金依頼されれば、その場で制限解除を行う。

[last]
L1 C CHGINI "@WORK@¥INI.DEF" !INI.DEF ;INI.DEFファイルに記述されている
                                         内容に従ってユーザのファイルを書
                                         き変える。(CHGINIマンド)
```

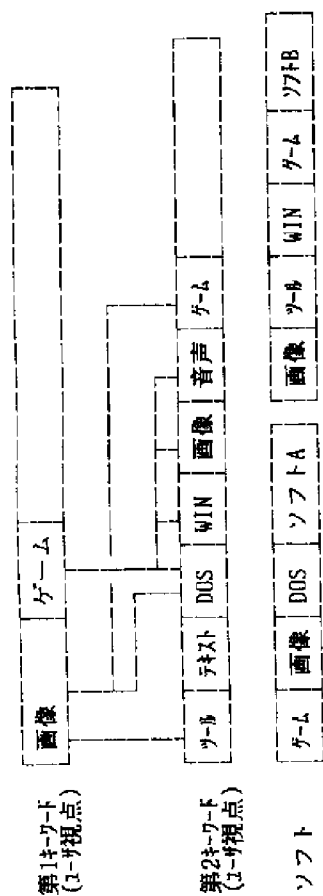
[Drawing 27]

価格決定処理のフローチャート



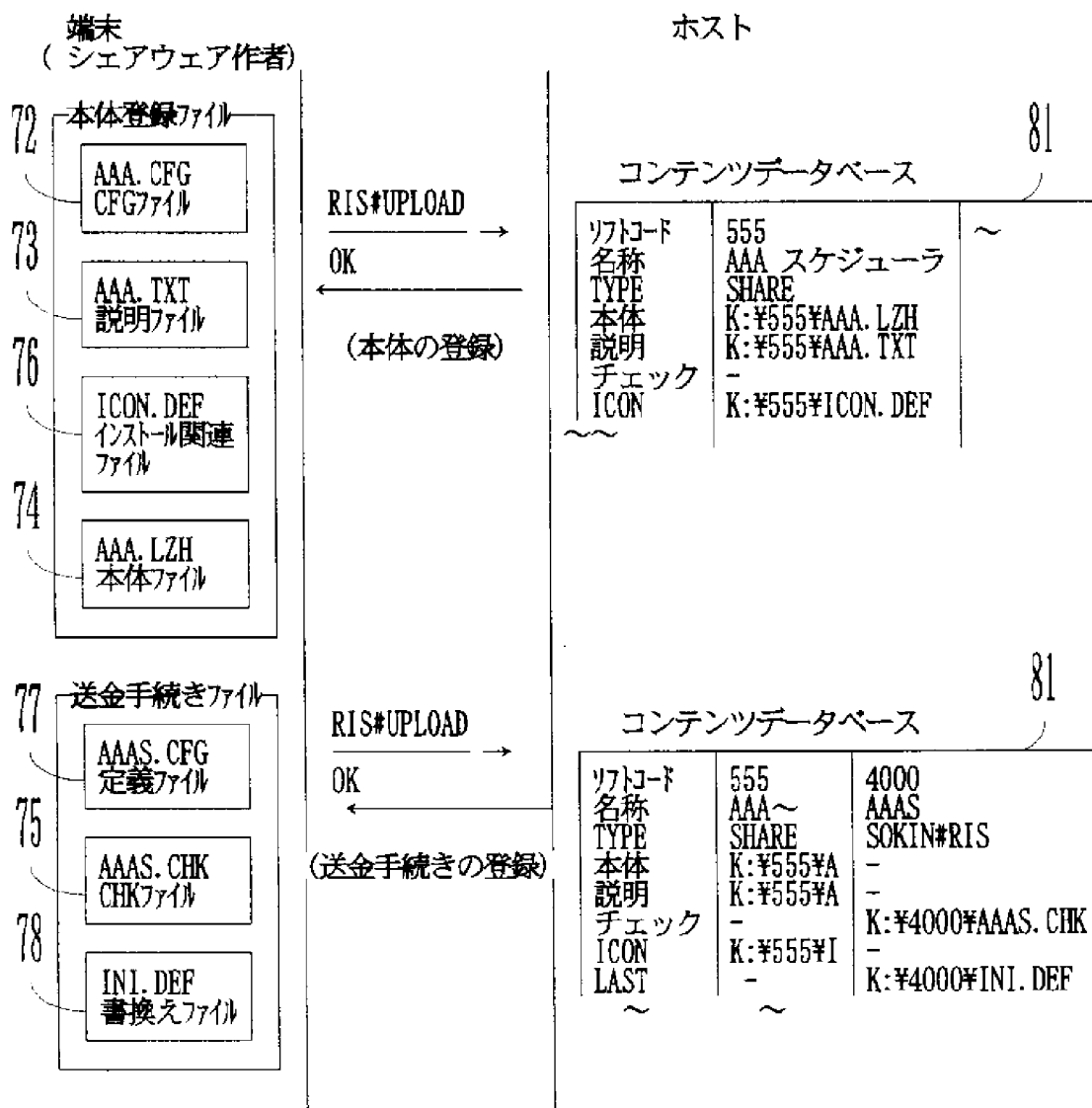
[Drawing 36]

キーワードの表示例を示す図



[Drawing 15]

アップロードを示す図



[Drawing 37]

環境情報の送信プロトコルを示す図

```

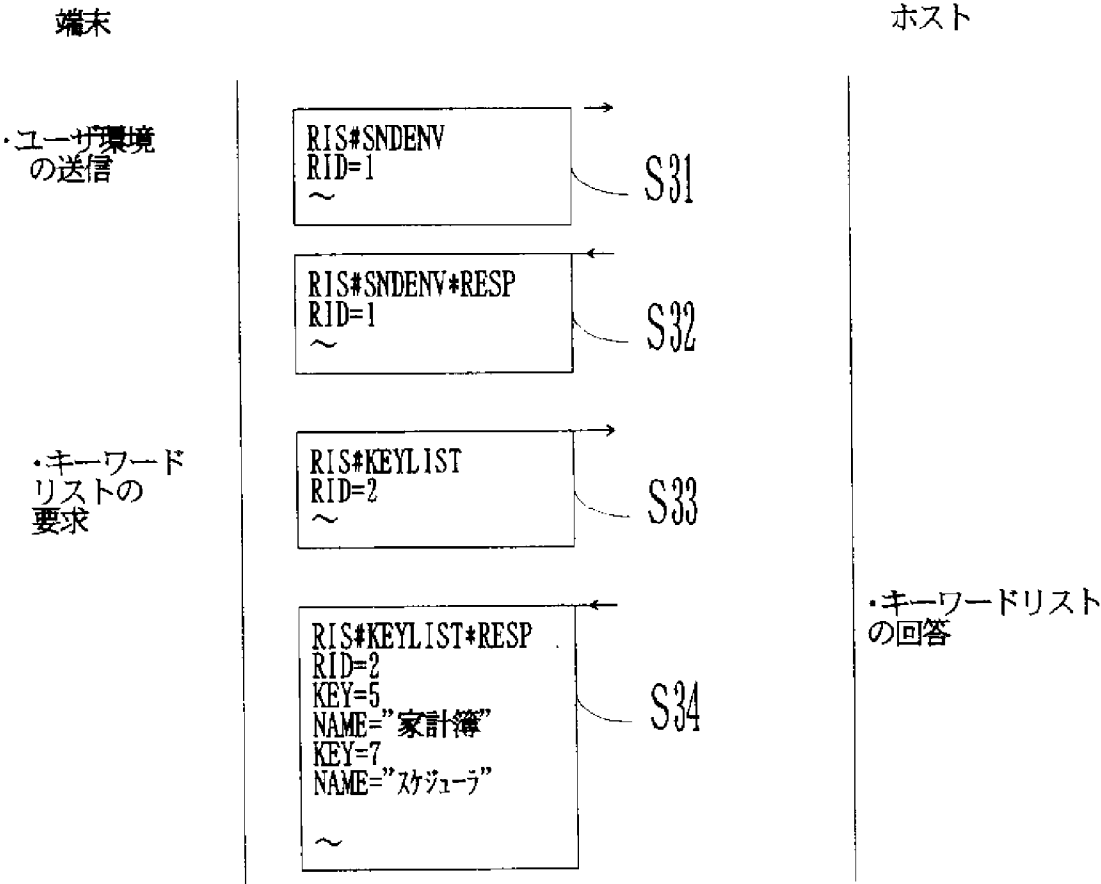
RIS_SENDENV
MACHINE:
MODEL=1, (端末ソフトのインストール時に取得)
OS=1,
DRV:
DRVKIND="HS",
HDINF:
HDID="0",
PARTINF:
PARTNO="1",
CAPACITY="50000000",
VACANT="50000000",
OSNAME="MSDOS",
DRVNAME="D:",
PRIORITY="1"

```

(ホストアクセス時に取得)

[Drawing 16]

シェアウェア手続きを示す図（その 1）



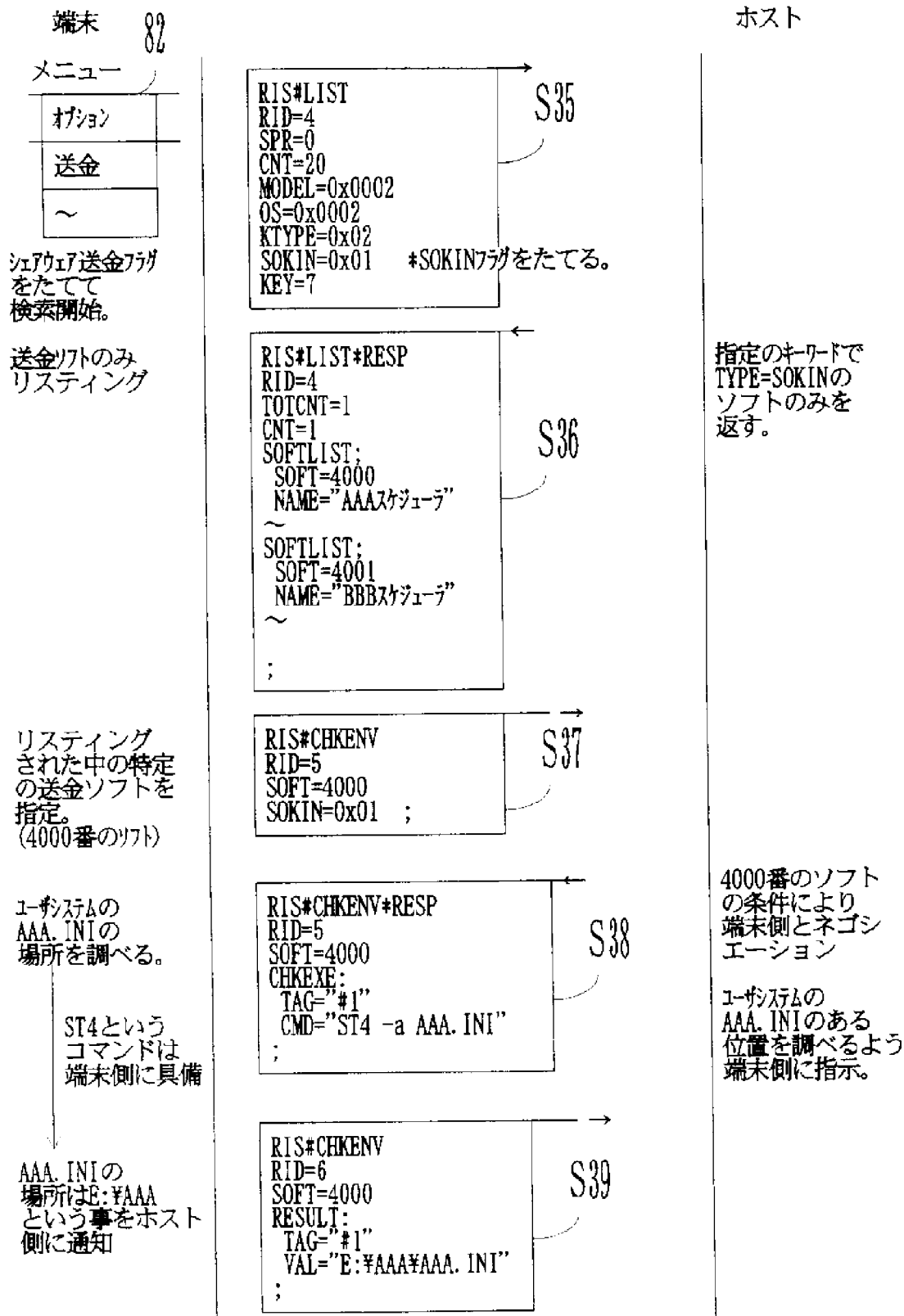
[Drawing 40]
設定ファイルの例を示す図

[DstDirs]
1=D:¥RIS¥KOBUTA
2=D:¥WINDOWS¥SYSTEM

[Files] ;Description,	Dst(Dir:Name),	Shared
Soft2.Exe, TOWNS.DRV	1:Soft2.Exe, 2:FONT.DRV	N N

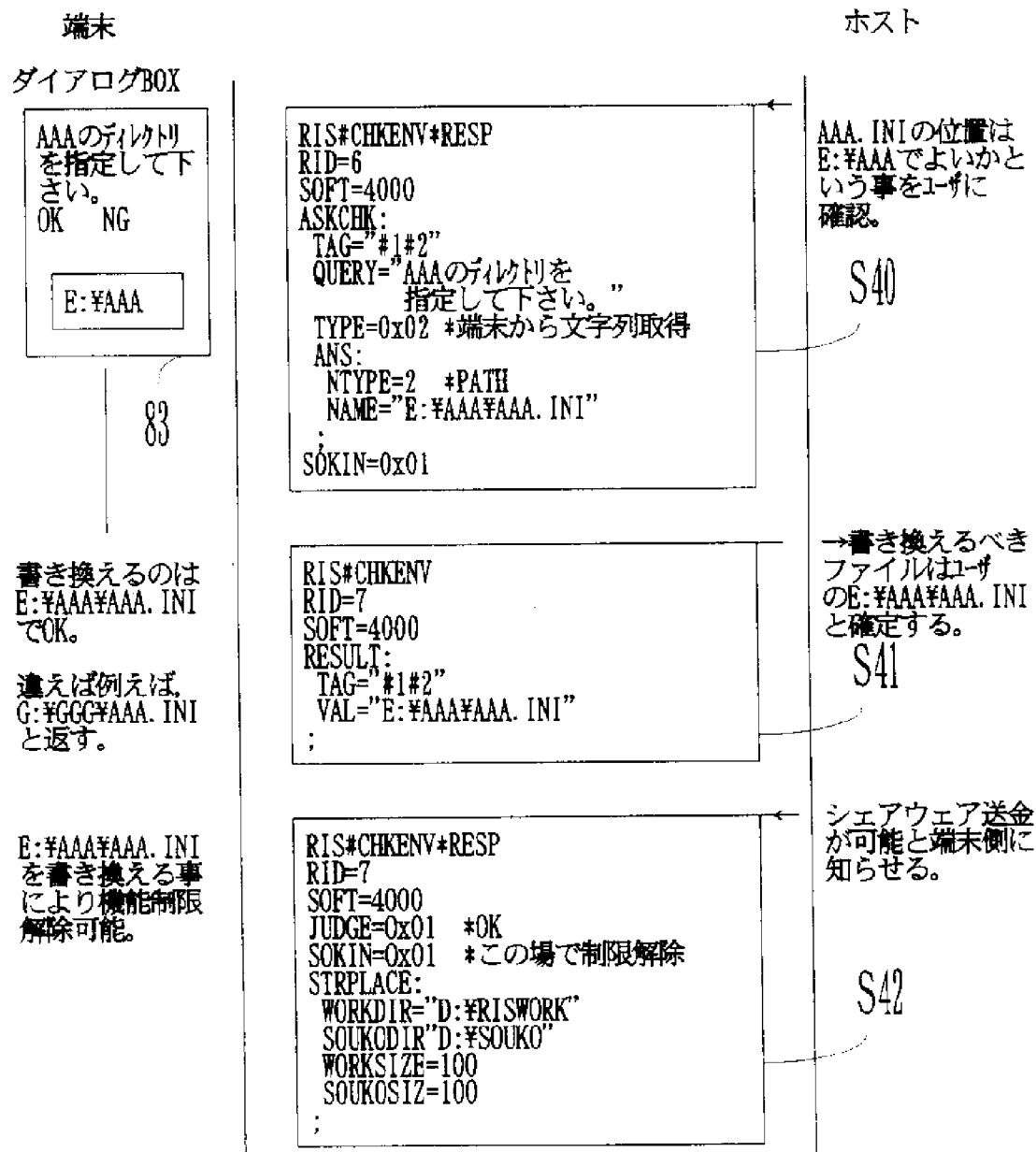
[Drawing 17]

シェアウェア手続きを示す図 (その2)



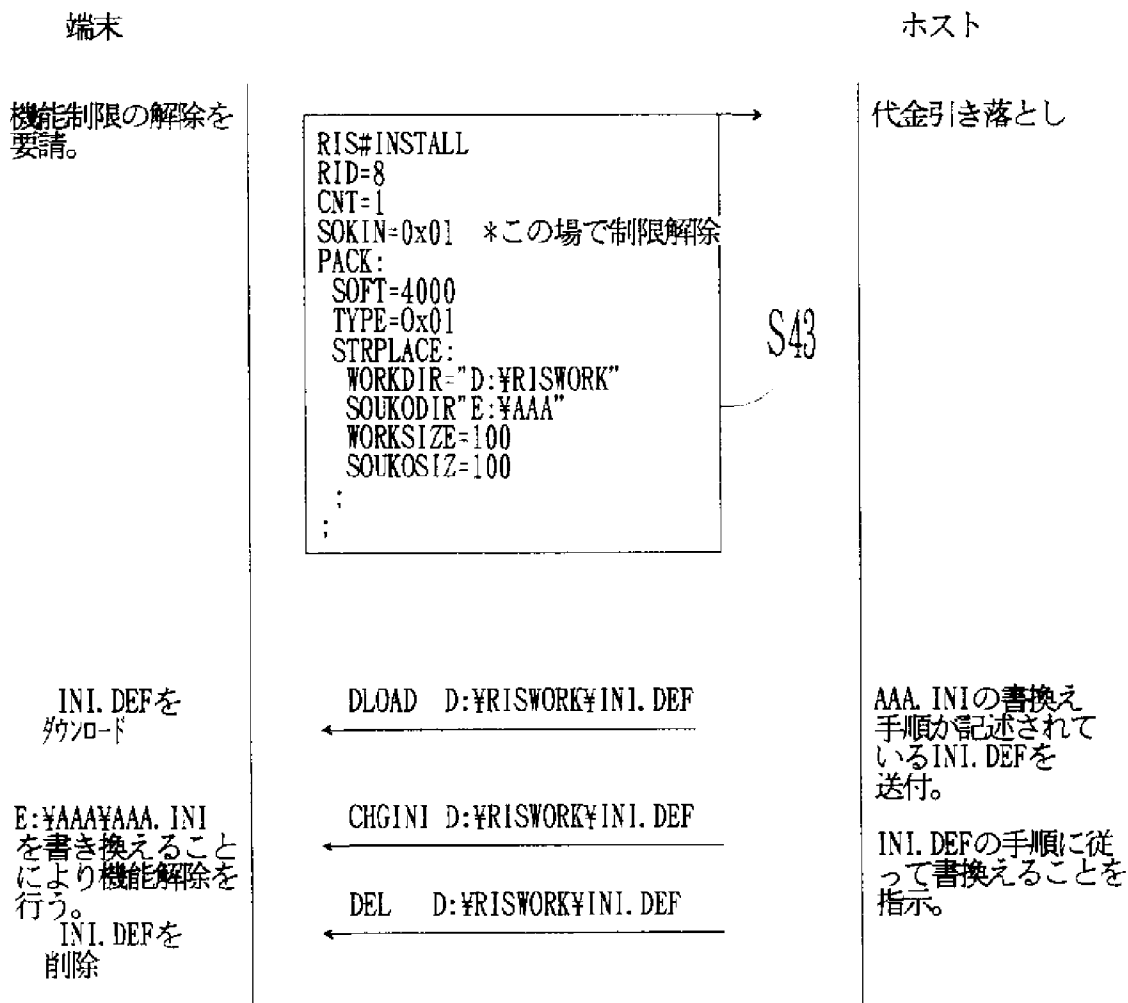
[Drawing 18]

シェアウェア手続きを示す図（その３）



[Drawing 19]

シェアウェア手続きを示す図（その４）



[Drawing 21]

第 2 の定義ファイルを示す図

定義ファイル—BBBS. CFG—

```

[name]
"BBBスケジューラ","ビースリー"

[type]
SOKIN 555 FJOKI BBB

[price]
4000

[machine]
TOWNS
FMR
PC98
PC-AT

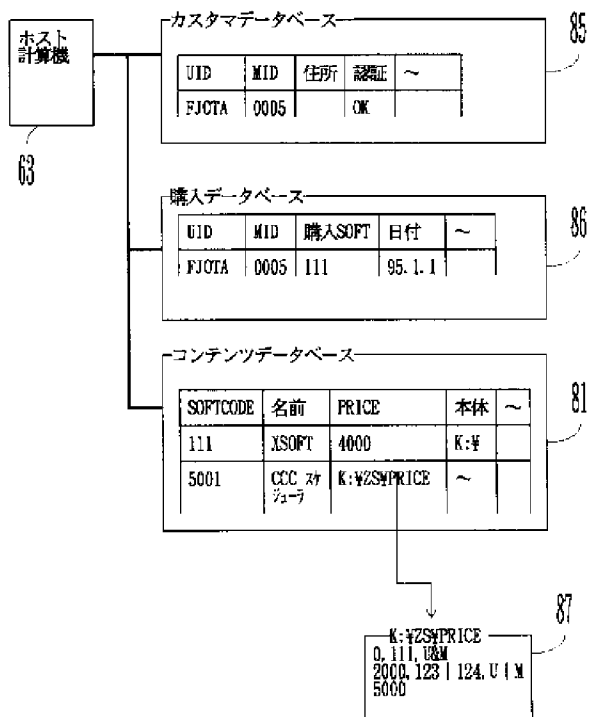
[os]
WIN31J

[key]
"T実用ソフト","Tスケジューラ"

[instype]
SOKIN#MAIL ;送金処理のみでその場での制限解除は行わない
  
```

[Drawing 26]

ホスト計算機のデータベースを示す図



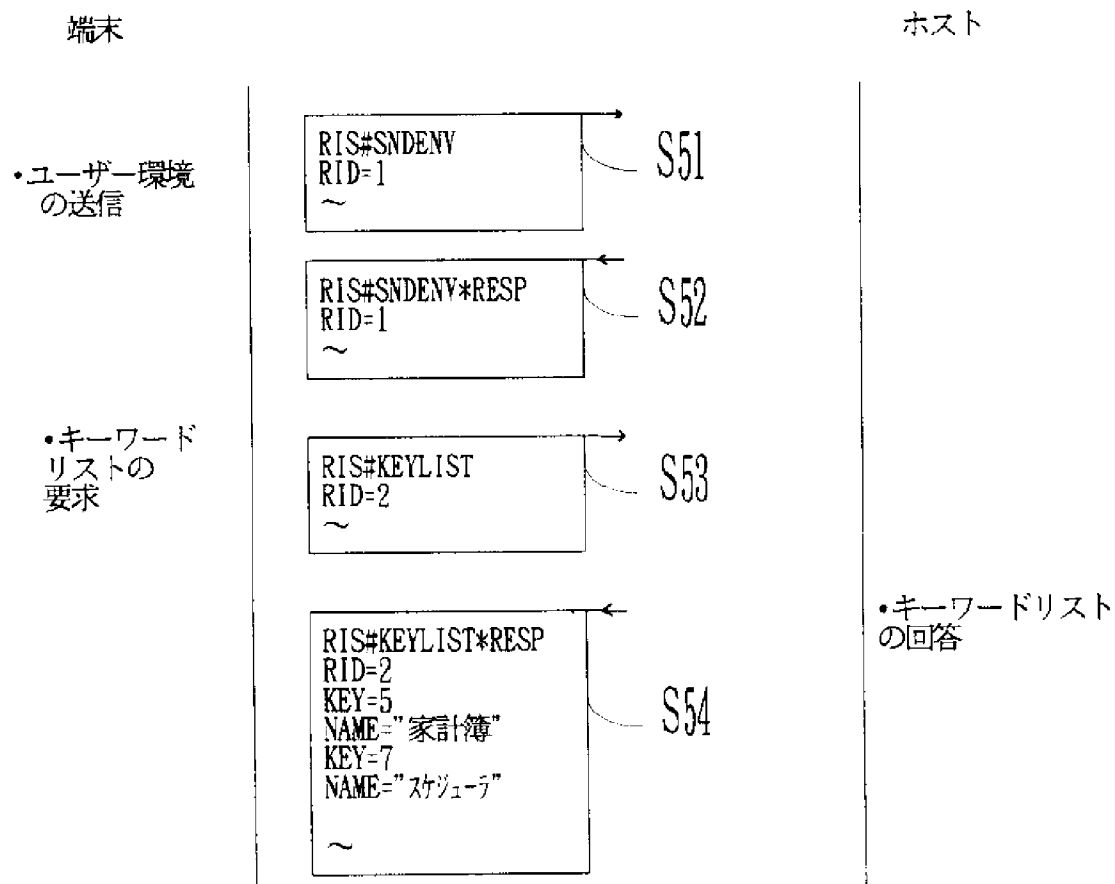
[Drawing 35]

キーワード選択のプロトコルを示す図

(コマンド)	RIS_KEYLIST
(レスポンス)	RIS_KEYLIST*RESP KEYLIST: KEY=1, NAME="OS/基本ソフト", KEY=2, NAME="開発支援", KEY=3, NAME="ゲーム", ~
(コマンド)	RIS_KEYLIST KEY=3
(レスポンス)	RIS_KEYLIST*RESP KEYLIST: KEY=51, NAME="RPG", KEY=52, NAME="アクション", KEY=53, NAME="パズル/クイズ", KEY=54, NAME="シミュレート", KEY=55, NAME="冗談", ~
(コマンド)	RIS_LIST KEY=52, KEY=53

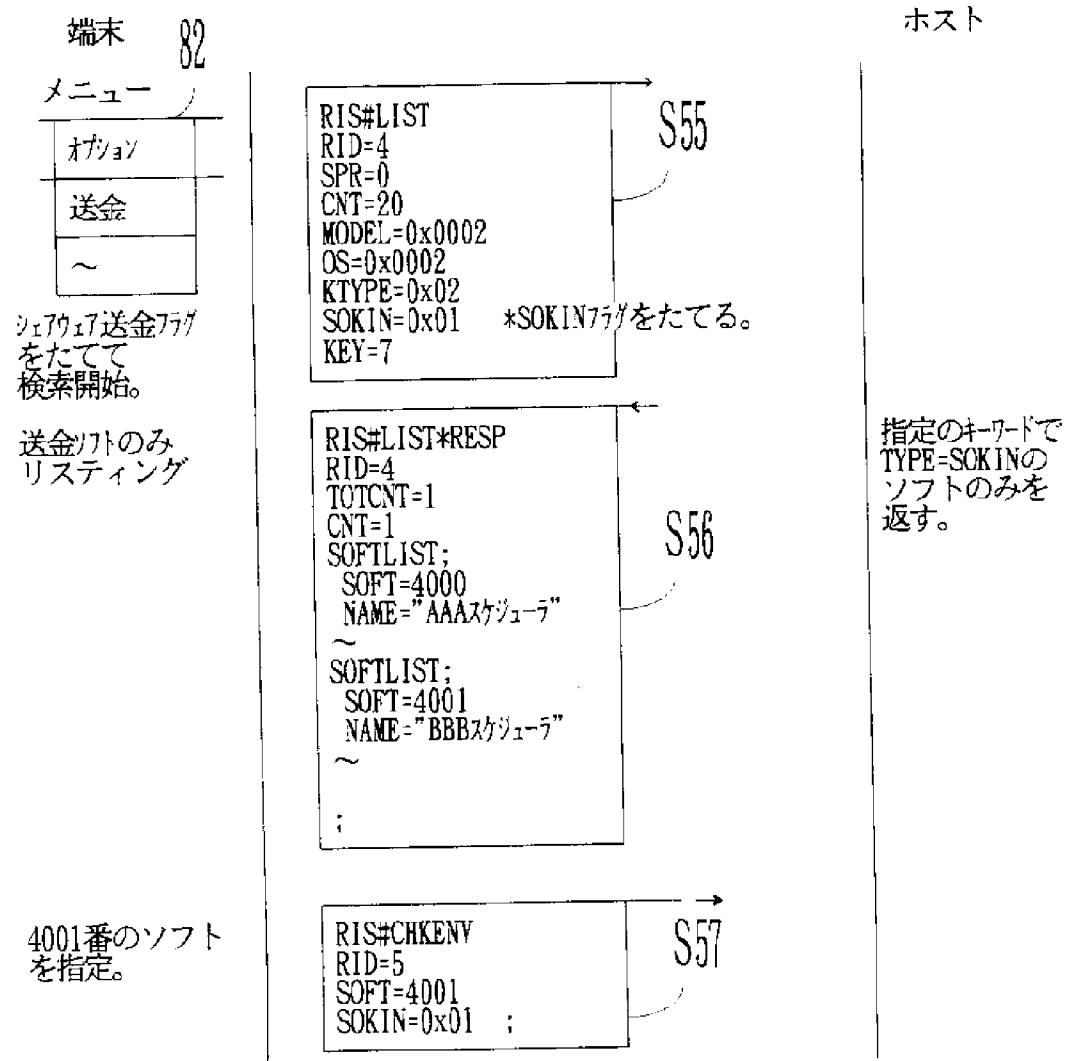
[Drawing 22]

代金引き落とし手続きを示す図（その１）



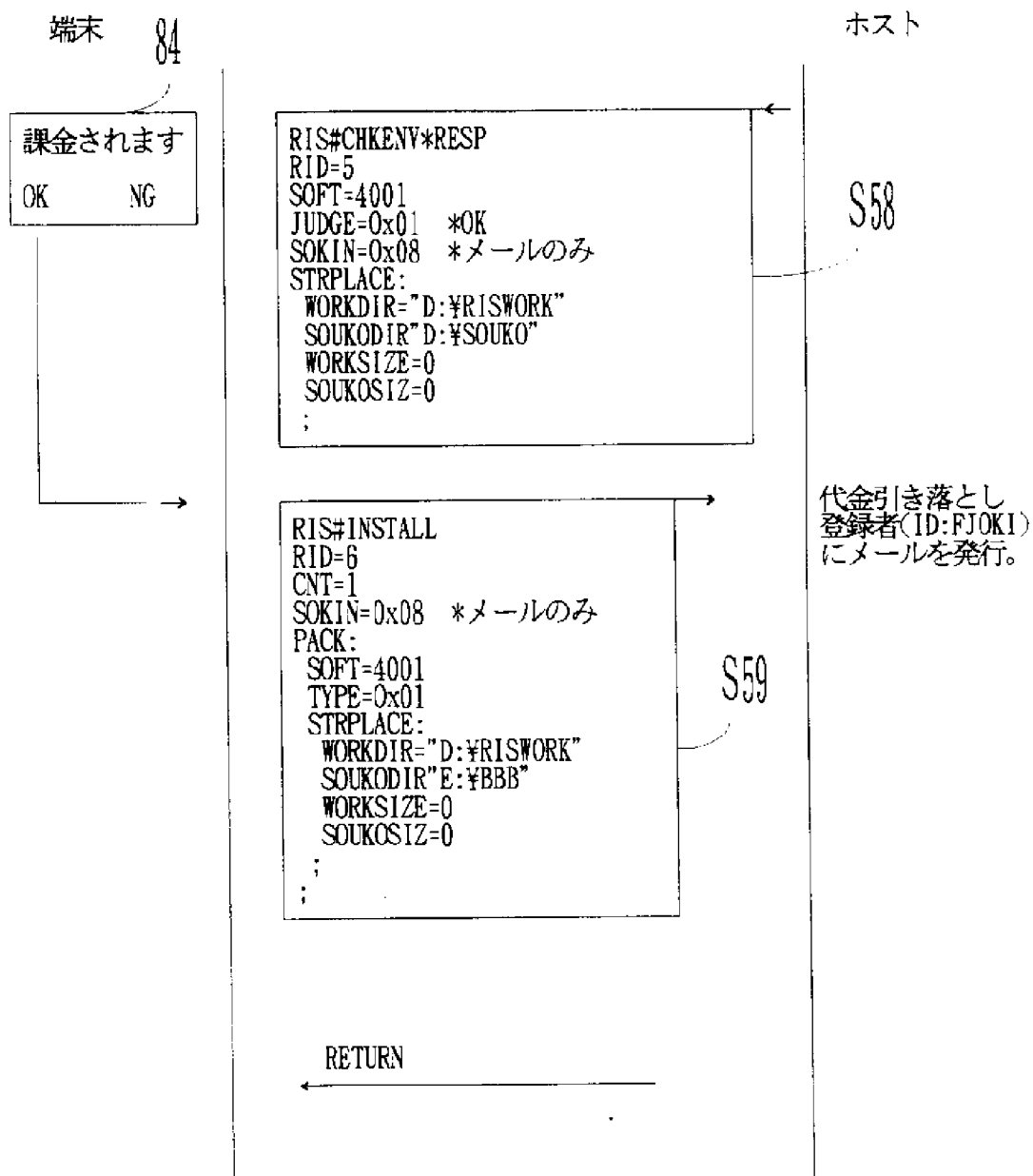
[Drawing 23]

代金引き落とし手続きを示す図 (その2)



[Drawing 24]

代金引き落とし手続きを示す図（その３）



[Drawing 25]

第3の定義ファイルを示す図

— 定義ファイル—CCC. CFG —

[name]
"CCCスケジューラ", "シースリー"

[type]
ISV

[price]
0 111 U&M
2000 123|124 U|M
5000

[machine]
TOWNS
FMR
PC98
PC-AT

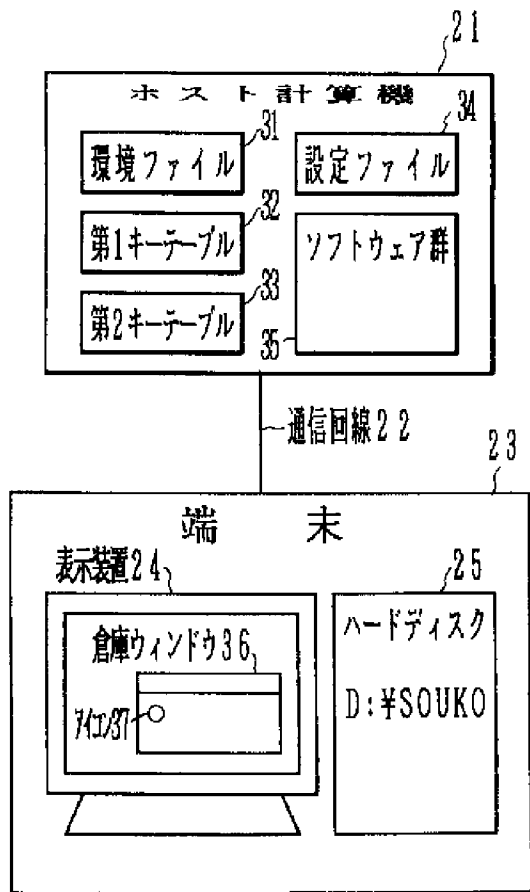
[os]
WIN31J

[key]
"T実用ソフト", "Tスケジューラ"

~

[Drawing 28]

先願のリモートインストールシステムの構成図

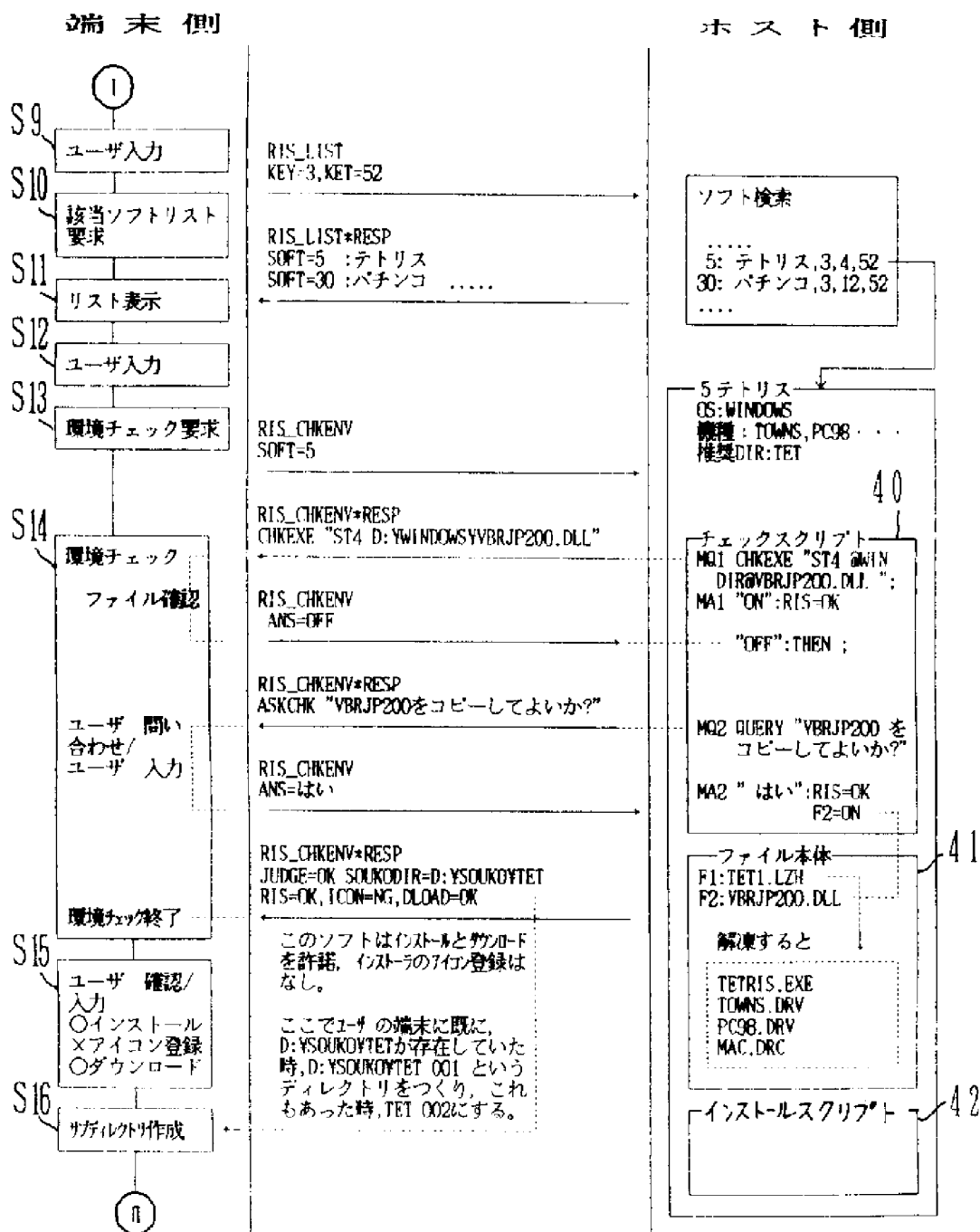


[Drawing 29]

端末側

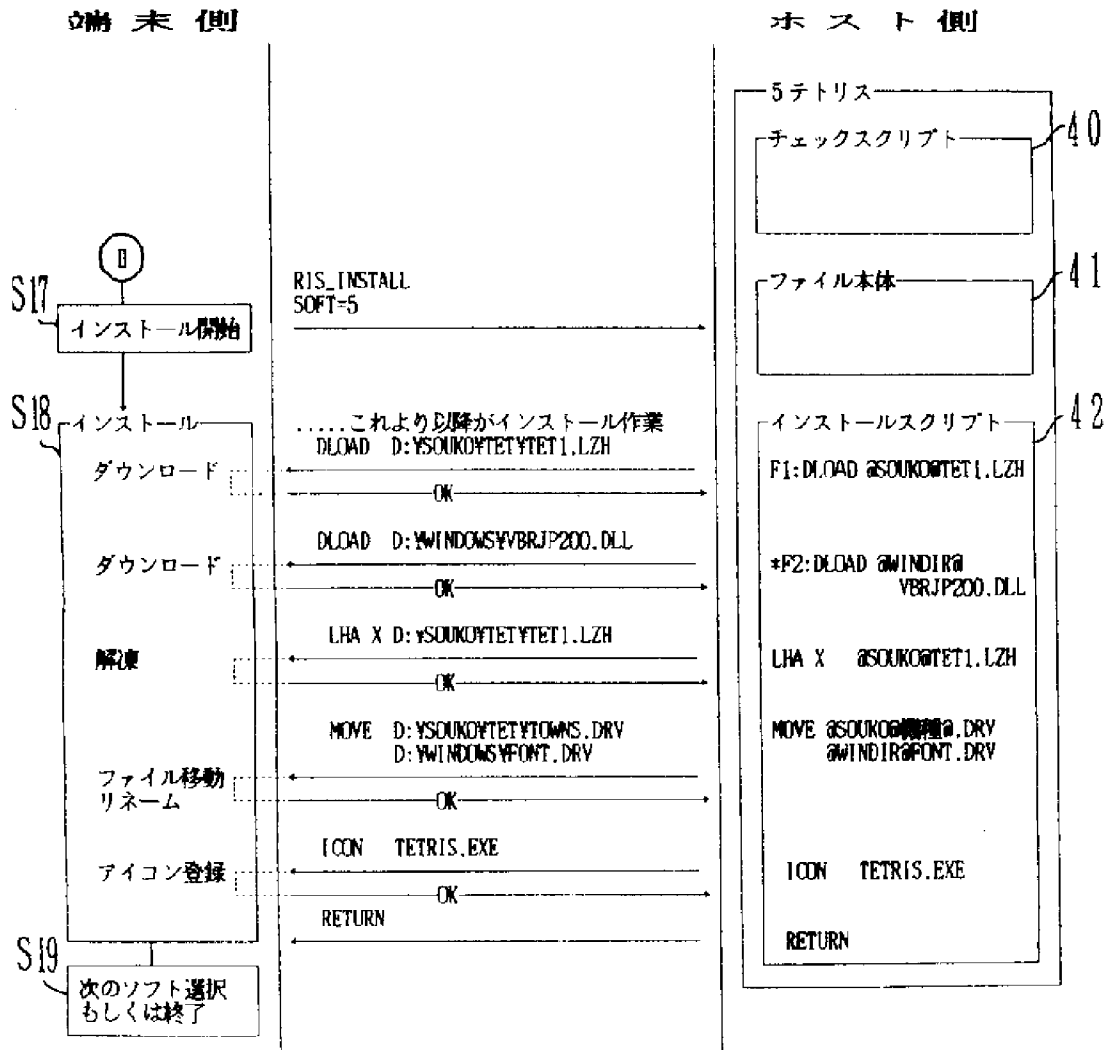


リモートインストールのフローチャート(その2)



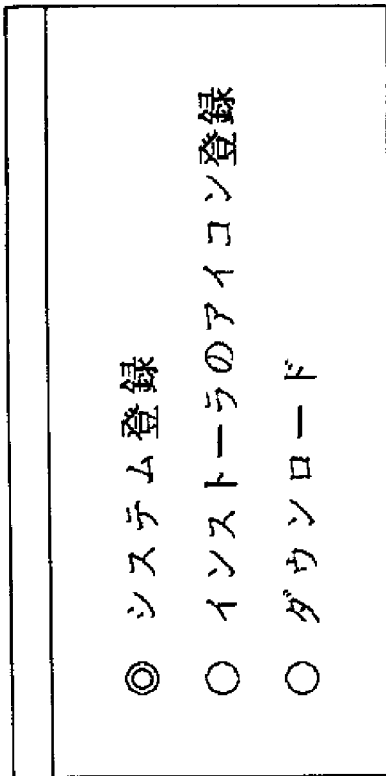
[Drawing 31]

リモートインストールのフローチャート(その3)



[Drawing 32]

インストール方法選択ウィンドウの例を示す図



[Drawing 33]

リモートインストールプロトコルを示す図(その1)

RIS_SENDEV(端末→ホスト)

```

RTD=1,
MID=1,
TIME=1994/08/94/12/20/20,
MACHINE:
  MODEL=1,
  OS=1,
  DRV:
    DRVKIND=HS,
    HDINF:
      HDID=0,
      PARTINF:
        PARTNO=1,
        CAPACITY=400000,
        VACANT=200000,
        OSNAME=MSDOS2,
        DRVNAME=D,
        PRIORITY=?
      ;
    PARTINF:
      PARTNO=2,
      CAPACITY=500000,
      VACANT=300000,
      OSNAME=MSDOS,
      DRVNAME=E,
      PRIORITY=?
    ;
  ;
  DRVKIND=HS,
  HDINF:
    HDID=1,
    PARTINF:
      PARTNO=1,
      CAPACITY=500000,
      VACANT=200000,
      OSNAME=MSDOS,
      DRVNAME=F,
      PRIORITY=?
    ;
  PARTINF:
    PARTNO=2,
    CAPACITY=700000,
    VACANT=300000,
    OSNAME=MSDOS,
    DRVNAME=G,
    PRIORITY=?
  ;
  ;
  DRVKIND=F2,
  DRVKIND=F2,
  DRVKIND=C
  ;
  SOUKODIR=D:\RISYSOUKO, <-----ここで倉庫ディレクトリを指定
  WINDIR=D:\WINDOWS,
  MEM
    MEM=8000000,
    TPA=400000
  ;
  ;
  MPSWD="1234"

```

~

[Drawing 38]

環境チェックのフローを示す図

RIS_CHKENV

SOFT=5 (ソフトコード5番をチェックしたい)

RIS_CHKENV*RESP

CHKEXE:

TAG="VBRJP200.DLL" (VBRJP200.DLLというファイルがシステムディレク
TORIにあるか調査); WORKDIR="D:\RIS\KOBUTA"
SOUKODIR="D:\RIS\KOBUTA"; WORKSIZ=700000,
SOUKOSIZ=500000

RIS_CHKENV

SOFT=5,

RESULT:

TAG="VBRJP200.DLL", (VBRJP200.DLLというファイルはシステムディレクトリにない)
VAL="OFF"

;

RIS_CHKENV*RESP

ASKCHECK:

TAG="Q1", (ユーザに問いあわせる)
QUERY="このソフトを実行するためには、VBRJP200.DLLが必要ですが、
あなたのシステムにはこのファイルが見つかりません。
このままでは実行できませんが、インストールしてもよろしいですか？",

TYPE=0,

ANS:

CODE="OK",
NAME="OK"

;

ANS

CODE="CANCEL",
NAME="CANCEL"

;

RIS_CHKENV

SOFT=5,

RESULT:

TAG="Q1", (ユーザがインストールしてもよいと答える)
VAL="OK"

RIS_CHKENV*RESP

JUDGE="OK" (インストール可能)

RIS_INSTALL

CNT=1,

TYPE="RIS",

STRPLACE:

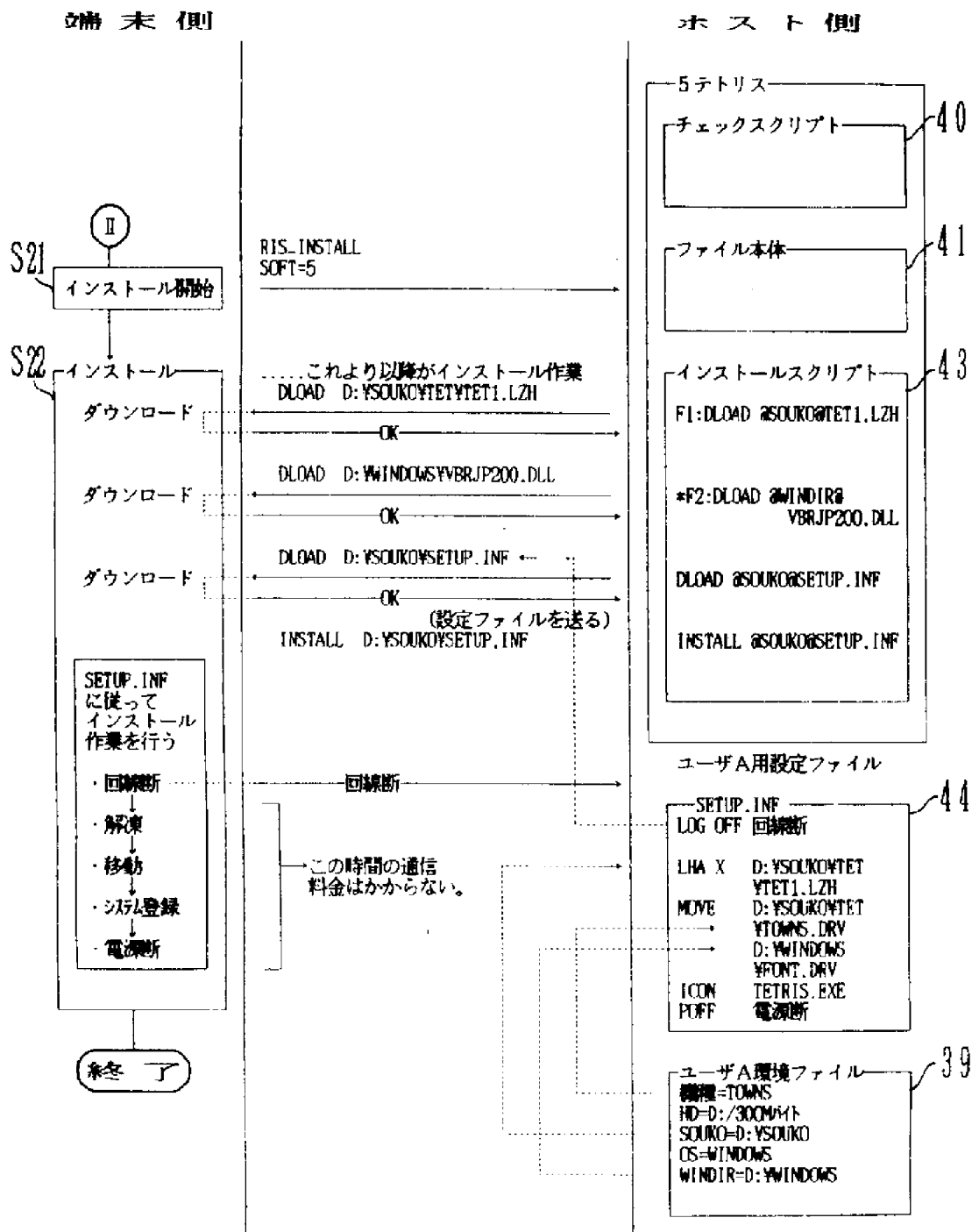
SOFT=5,

WORKDIR="D:\RIS\KOBUTA",
SOUKODIR="D:\RIS\KOBUTA"

;

[Drawing 39]

自動インストールのフローチャート



[Translation done.]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the service which installs software in a user's terminal unit through a communication line, and relates to the software price sanction system which sanctions the price of software, and its approach.

[0002]

[Description of the Prior Art] Although the hope of the user who wants for software to come to hand through a communication line has arisen with development of personal computer communications in recent years etc., the full-scale system which delivers such software automatically is not realized. In order to perform automatic delivery of software, a delivery center is told about the software which a user wishes first from the terminal unit of a house, and the delivery center which received the notice from the user can consider the format of transmitting the software demanded through the communication line to a user's terminal unit.

[0003] In order to realize automatic delivery of such software, "the remote install system and approach" (Japanese Patent Application No. 7-1797) of point ** were devised. Hereafter, the technique of point ** is explained, referring to a drawing.

[0004] Drawing 28 is the remote install structure-of-a-system Fig. of point **. The remote install system of drawing 28 consists of the host computer 21 in the delivery center of software, a user's terminal 23, and the communication line 22 that connects these. A terminal 23 shall be a personal computer at a user's house distant distantly from a delivery center, and a communication line 22 shall be a circuit for the networks of personal computer communications.

[0005] The host computer 21 stores the 1st key 2nd key table 32 and 33 holding the list of keywords used when choosing specific software from the software group 35 containing two or more software which can be delivered in the storing field which is not illustrated, and its software group 35.

[0006] If a user demands the list of keywords of a host computer 21 from a terminal 23, a host computer 21 will carry out sequential transmission of the 1st key table 32 and the 2nd key table 33, and will display on the screen of the display 24 of a terminal 23 the keyword contained in them. A user chooses the thing corresponding to the software which he wishes from the displayed keyword, and notifies to a host computer 21.

[0007] A host computer 21 displays the menu containing the name of some software applicable to the notified keyword on the screen of a display 24, and a user chooses the software which he wishes out of it, and notifies it to a host computer 21. And a host computer 21 takes out the contents (file) of the software which the user chose from the software group 35, and stores them in directory D:\SOUKO for delivery (delivery or distribution) set as the hard disk 25 of a terminal 23.

[0008] At this time, the icon 37 for starting the delivered software is registered automatically, and is displayed in the warehouse window 36 prepared on the screen of an indicating equipment 24 corresponding

to directory D:\SOUKO. For example, the delivered software is registered into the program manager of WINDOWS when the terminal 23 carries WINDOWS. Henceforth, a user only clicks on an icon 37 using input units, such as a mouse, and can use the delivered software.

[0009] The environment file 31 which described the operating environment of a hardware, a software tool, etc. which a terminal 23 equips is generated within a terminal 23, and is sent to a host computer 21 at the time of access of a host computer 21. A host computer 21 holds the received environment file 31 for every user, and uses it for the check of whether the software which the user chose can operate.

[0010] Moreover, a host computer 21 prepares the configuration file 34 according to individual for every user, and describes to this a package of a file required for actuation and its storing location of the software which the user chose. And a configuration file 34 can be put in block to a terminal 23, and automatic install of software can be made to carry out to delivery and a terminal 23 with those packages of a file.

[0011] Next, the flow of delivery of the software by the remote install system of drawing 28 of operation is explained, referring to from drawing 29 to drawing 31. In drawing 29, User's A terminal 23 creates the environment file 38 which set up the information which acquired and acquired the information on operating environment at the time of install of the terminal software for a communication link (step S1). At this time, the information which time amount requires for acquisition of the storing location SOUKO used for the model of a user's terminal or delivery, or the information which a user must be asked by the case is acquired.

[0012] In the decision of the storing location SOUKO, it investigates whether a terminal 23 has a free area more than a predetermined capacity in a hard disk 25, and if there is a free area, the directory for delivery will be created in the root. At this time, a terminal 23 generates a directory name etc. automatically, and User A does only the activity which checks it. Therefore, User A does not need to input a directory name etc.

[0013] Here, user's A model is TOWNS and it is written in an environment file 38 that the directory of SOUKO is D:\SOUKO (the directory SOUKO of drive D). If User A has the need, he can also change D:\SOUKO into other directories.

[0014] If there is no free area of a predetermined capacity in the already set-up partition, the free area of another partition will look for the largest location, and will create the directory for delivery there. Supposing directory D:\SOUKO fills, specifically, "D:\SOUKO is full of a terminal 23. A warehouse is changed into F:\SOUKO. Is it all right? Messages, such as ", are displayed on the screen of a display 24. If User A recognizes this, F:\SOUKO will newly serve as a directory of SOUKO. the time of there being no free area of a predetermined capacity in every hard disk -- " -- disk capacity is lacking though regrettable. Please extend a disk. Messages, such as ", are displayed.

[0015] Next, the information which may have changed after install of a hard disk, the situation of memory, etc. is acquired at the time of starting of terminal software (at the time of access to a host computer 21) (step S2). Here, User's A hard disk is in drive D, and it is written in an environment file 38 that an availability is 300 M bytes. In this way, the contents of the created environment file 38 are transmitted to a host computer 21 by Command RIS SENDENV, when User A accesses a host computer 21 (connection) (step S3).

[0016] A host computer 21 holds the received information as a user A environment file 39. OS (operating system) currently used and its storing location are described besides a model, hard-disk-information HD, and the storing location SOUKO by the user A environment file 39. Here, OS of User's A terminal 23 is WINDOWS, and it turns out that the storing location WINDIR is D:\WINDOWS.

[0017] It is RIS SENDENV*RESP as a response to the command RIS SENDENV from a host computer 21. If O.K. is received, a terminal 23 will require the 1st key list with Command RIS KEYLIST (step S4).

According to this, a host computer 21 returns the contents of the 1st key table 32 with RIS KEYLIST*RESP. Here, OS/operating system, development exchange, a game, and ... are stored in the 1st

key table 32 as a keyword corresponding to the key numbers 1, 2, and 3 and ...

[0018] If these keywords are displayed on the screen of a display 24 as the 1st key list (step S5), User A will choose the 1st keyword from them, and will input into a terminal 23 (step S6). Then, a terminal 23 sends the command RIS KEYLIST which requires the 2nd key list with the key number of the 1st keyword which User A chose to a host computer 21 (step S7). Here, User A chooses a game as the 1st keyword, and the key number 3 corresponding to it is sent to a host computer 21.

[0019] The host computer 21 of which the 2nd key list was required asks for the corresponding 2nd key table 33 using the pointer stored in the 1st key table 32 corresponding to the received key number, and returns the contents with RIS KEYLIST*RESP. Here, RPG, action, a puzzle/quiz, and ... are stored in the 2nd key table 33 as a keyword corresponding to the key numbers 51, 52, and 53 and ...

[0020] Generally two or more 2nd key tables are prepared corresponding to the keyword in the 1st key table 32, and the number is the same as the number of the keywords in the 1st key table 32, or there is than it. [few] In the case of the latter, two or more keywords in the 1st key table 32 will point out the same 2nd key table.

[0021] If the keyword in the 2nd key table is displayed on the screen of a display 24 as the 2nd key list (step S8), User A will choose the 2nd keyword from them, and will input into a terminal 23 (drawing 30 , step S9). Then, a terminal 23 sends the command RIS LIST which requires the list of the software which corresponds to both the 1st and the 2nd keyword with the 1st and the key number of the 2nd keyword which User A chose to a host computer 21 (step S10). Here, User A chooses action as the 2nd keyword, and the key number 52 corresponding to it is sent to a host computer 21.

[0022] The host computer 21 of which the list of software was required searches software with the 1st and two key numbers of the 2nd keyword out of the software group 35. At this time, it searches for a flat, without distinguishing the 1st keyword and the 2nd keyword as retrieval conditions. Moreover, the classification of a model or OS is treated as a default key, and after also considering these, it is searched. Thereby, it is prevented that the software only for example, other than TOWNS and for models is searched.

[0023] And the corresponding name of software and the list of numbers are sent to a terminal 23 with RIS LIST*RESP. Here, since software with the key numbers 3 and 52, such as TETORISU and pachinko, corresponds, those names are sent to a terminal 23 with each software number 5 and 30 grades.

[0024] If the list of software is displayed on the screen of a display 24 (step S11), User A will choose the software which he wishes out of them, and will input into a terminal 23 (step S12). Then, a terminal 23 sends the command RIS CHKENV which requires a check with User's A environment suitable for actuation of the software with the number of the software which User A chose to a host computer 21 (step S13). Here, User A chooses TETORISU and the software number 5 corresponding to it is sent to a host computer 21.

[0025] The host computer 21 which received the number of the software which User A chose prepares the check script 40 for investigating the adjustment of the operating environment of software and the environment of User's A terminal 23 corresponding to the number, and performs an environmental check. Since this check is automatically performed by the exchange between the executive program of the check script 40, and the terminal software of a terminal 23, User A does not necessarily need to be conscious of the environmental check being performed (step S14). Only when it is necessary to perform a certain inquiry to User A, a host computer 21 performs the inquiry.

[0026] Here, it is described as operating environment of TETORISU which User A chose that WINDOWS and a model are [recommendation directory (DIR) names, such as TOWNS and PC98,] TET(s) for OS. On the other hand, it turns out that a model is described to be TOWNS, OS is described to be WINDOWS by the user A environment file 39, and the model and OS conform to it by comparing both.

[0027] next -- TETORISU -- a check -- a script -- 40 -- seeing -- if -- a user -- A -- a side -- storing -- a location -- WINDIR -- VBRJP -- 200 . -- DLL -- ** -- saying -- a file -- it is -- a ***** -- investigating

-- a sake -- a command -- " -- ST -- four -- -- @ -- WINDIR@VBRJP -- 200 . -- DLL -- " -- it is -- since -- (MQ1) a host computer 21 -- this -- RIS CHKENV*RESP -- a terminal 23 -- sending . At this time, a host computer 21 replaces and sends @WINDIR@ to D:\WINDOWS with reference to the user A environment file 39. Here, @ expresses a wild card. Moreover, file VBRJP200.DLL is one of the files required for actuation of TETORISU.

[0028] The terminal 23 which received this command investigates whether file VBRJP200.DLL is in the directory WINDOWS of drive D, and returns it to a host computer 21 by setting that result to ANS. Here, since there was no corresponding file, ANS=OFF is returned.

[0029] May the host computer 21 which got to know that there was no file VBRJP200.DLL copy "VBRJP200 to a terminal 23 according to the check script 40 (MQ2)? Delivery and this inquiry are displayed on a terminal 23 by the screen of a display 24 in the inquiry ". User A inputs the reply to the displayed inquiry, and a terminal 23 returns the reply to a host computer 21. Here, ANS= is. It is returned, a host computer 21 consents to remote install according to the check script 40 (RIS=OK), and the flag F2 which directs the copy of VBRJP200.DLL is set (MA2).

[0030] Since ANS=ON is returned when file VBRJP200.DLL is in the directory to which the terminal 23 was specified, it becomes RIS=OK at the time (MA1).

[0031] Thus, by performing an environmental check automatically, it can prevent delivering the software which does not suit User's A environment. for example, after purchasing a certain packaged software through a communication line 22, if there is no specific driver, it will know that it will not operate -- like -- accident is prevented beforehand.

[0032] When it comes to RIS=OK, a host computer 21 ends an environmental check and sends the directory SOUKODIR of the address for delivery to a terminal 23 with a judgment result (JUDGE=OK). This SOUKODIR is specified in the format which added as a subdirectory TET which is the recommendation directory of TETORISU to the bottom of D:\SOUKO which is the directory of SOUKO stored in the user A environment file.

[0033] At this time, the propriety (RIS) of install, the existence (ICON) of icon registration of an install program (installer), and the propriety (DLOAD) of download are sent to a terminal 23 at coincidence. These flags RIS, ICON, and DLOAD notify to a terminal 23 which [of install, icon registration of an installer, and the downloads] is possible for a host computer 21.

[0034] Install means registering into the system of a terminal 23, for example, WINDOWS, the software which User A chose, and making it usable on a terminal 23. Therefore, even the activity which carries out icon registration of the execution file of that software on WINDOWS in this case is included. On the other hand, icon registration of an installer means carrying out icon registration of the program which performs install on a terminal 23.

[0035] Here, install and download are permitted (RIS=OK, DLOAD=OK) and the conditions of not performing icon registration of an installer (ICON=NG) are shown. In the case of software with a complicated install program, the purport which needs icon registration of an installer is shown instead of install being permitted. Moreover, only download is permitted when the application for TOS (OS of TOWNS) is required from the terminal which carries WINDOWS.

[0036] Next, the terminal software of a terminal 23 gives priority in order of install, icon registration of an installer, and download, sets up as a default what has more high priority, and displays it on the screen of a display 24. Here, among install permitted by the host computer 21, and download, higher install of priority is set up as a default and displayed on the install approach selection window.

[0037] Drawing 32 shows the example of a display of the install approach selection window. In drawing 32, "system registration" is equivalent to install and this is chosen by the default.

[0038] User A checks the displayed install approach and inputs the checked purport (step S15). Moreover, User A can also change a setup displayed here. For example, "icon registration of an installer" in the install

approach selection window of drawing 32 is chosen and inputted to perform icon registration of an installer. [0039] "System registration" is chosen to perform install which User A can do it fundamentally, without applying time and effort, and suits, "icon registration of an installer" is chosen to perform a fine install setup by oneself, and "download" is chosen to change a storing location later (I want to install in the terminal of another model). If "download" is chosen, it will also become possible to find out whether different software for models from the model of terminal 23 comes to hand, and it operates.

[0040] Next, a terminal 23 generates automatically subdirectory D:\SOUKOTET for delivery directed from the host computer 21 in a hard disk 25 (step S16). It is D:\SOUKOTET, when it carries out also here and subdirectory D:\SOUKOTET has already existed in the terminal 23. It is D:\SOUKOTET, when the subdirectory 001 is built and this has also already existed. The subdirectory 002 is built.

[0041] The file body 41 of TETORISU consisted of file TET1.LZH (F1) and VBRJP200.DLL (F2), and TET1.LZH compressed four file TETRIS.EXE(s) and TOWNS.DRV(s), PC98.DRV, and MAC.DRC, and has done them. If it elongates in the condition before compressing TET1.LZH (defrosting), it will be divided into these four files, but defrosting of TET1.LZH is performed after being delivered by the terminal 23 from a host computer 21.

[0042] The terminal 23 which generated the subdirectory for delivery is sent to a host computer 21 with the number of the software which chose the command RIS INSTALL which requests initiation of remote install (drawing 31 , step S17). In response, a host computer 21 starts remote install of the software corresponding to the sent number. Remote install is automatically performed by the exchange between a host computer 21 and a terminal 23 according to the installation script 42 of TETORISU which the host computer 21 created (step S18).

[0043] An installation script 42 has the description which directs to download file TET1.LZH to storing location @SOUKO@ by the side of User A first. Then, a host computer 21 transposes @SOUKO@ to SOUKODIR=D:\SOUKOTET, and downloads TET1.LZH to subdirectory D:\SOUKOTET of a hard disk 25.

[0044] If completion (O.K.) of download is notified from a terminal 23, next, a host computer 21 will transpose @WINDIR@ to D:\WINDOWS, and will download VBRJP200.DLL to directory D:\WINDOWS of a hard disk 25.

[0045] When completion (O.K.) of download is notified from a terminal 23, a host computer 21 is the directions and LHA which thaw TET1.LZH downloaded to storing location @SOUKO@ (D:\SOUKOTET) next. X D:\SOUKOTET\TET1.LZH is sent. In response, a terminal 23 is thawed to four file TETRIS.EXE(s) and TOWNS.DRV which mentioned TET1.LZH above, PC98.DRV, and MAC.DRC. These four files are held at the same subdirectory D:\SOUKOTET as TET1.LZH.

[0046] When completion (O.K.) of defrosting is notified from a terminal 23, a host computer 21 is the directions and MOVE which are made to move a file called model @.DRV of storing location @SOUKO@ (D:\SOUKOTET) to storing location @WINDIR@ (D:\WINDOWS), and change a file name into FONT.DRV next. D:\SOUKOTET\TOWNS.DRV D:\WINDOWS\FONT.DRV is sent. At this time, a host computer 21 replaces and sends model @ to TOWNS with reference to the user A environment file 39. In response, a terminal 23 moves file TOWNS.DRV of subdirectory D:\SOUKOTET to directory D:\WINDOWS (file migration), and changes it into the file name of FONT.DRV (renaming).

[0047] When completion (O.K.) of file migration and renaming is notified from a terminal 23, a host computer 21 is the directions and ICON which perform icon registration of file TETRIS.EXE next. TETRIS.EXE is sent. In response, a terminal 23 iconifies file TETRIS.EXE of subdirectory D:\SOUKOTET, and registers it into a terminal 23. The icon 37 which starts TETRIS.EXE is displayed by this in the warehouse window 36 displayed on the screen of a display 24, and if it clicks on an icon 37, TETORISU will start actuation.

[0048] If completion (O.K.) of icon registration is notified from a terminal 23, a host computer 21 will

return RETURN, will notify termination of remote install to a terminal 23, and will end a series of install activities. The terminal 23 to which termination of remote install was notified performs selection and its remote install of the following software according to directions of User A, or ends processing (step S19). [0049] At the time of install of step S18, if software UEEA to install is large compared with the availability of the storing location of the download place, a storing location will be changed and downloaded.

[0050] Next, the example of the protocol of remote install and the example of a display of the 1st and 2nd keywords are explained, referring to from drawing 33 to drawing 38. Drawing 33 shows an example of the protocol with which a terminal 23 sends environmental information at the time of access to the host computer 21 by the terminal 23 of step S3. the machine ID (MID) which specifies Request ID (RID) and a terminal in the protocol of drawing 33, and time (TIME) -- then, the machine information (MACHINE:) about a terminal 23 is described.

[0051] The information about the hard disks 25, such as capacity and a drive name, is described by the drive information in MACHINE: (DRV:). For example, CAPACITY in PARTINF: expresses the capacity of the partition, VACANT expresses the availability of them and DRVNAME expresses the drive name corresponding to the partition.

[0052] DRV: Then, the directory WINDIR which is the storing location of the warehouse directories SOUKODIR and OS (WINDOWS) which are the storing locations for delivery is described. Here, it is SOUKODIR=D:\RIS\SOUKO and is WINDIR=D:\WINDOWS. There is information (MEM:) about memory in the degree. By specifying this SOUKODIR, a terminal 23 notifies a delivery place to a host computer 21.

[0053] And MACHINE: Then, the machine password (MPSWD) of a terminal 23 etc. is described. Drawing 34 shows an example of the protocol with which a host computer 21 adds and returns a subdirectory to SOUKODIR at the time of termination of the environmental check of step S14. the protocol of drawing 34 - setting -- RID and the judgment result (JUDGE) of an environmental check -- then, the information (STRPLACE:) about a delivery place is described.

[0054] SOFT in STRPLACE: expresses with step S12 the number of the software which the user chose, WORKDIR expresses a working area, and SOUKODIR expresses the delivery place specified by a host computer 21. Here, WORKDIR is the same as SOUKODIR of drawing 33, and SOUKODIR has a form which added subdirectory name FM to SOUKODIR of drawing 33.

[0055] STRPLACE: Then, the size WORKSIZ of a working area and the size SOUKOSIZ of a delivery place are described. Drawing 35 shows an example of the protocol with which a terminal 23 returns the key number of the keyword from which delivery and User A chose [the host computer 21] the 1st and 2nd key list in S9 at the terminal 23 from step S4. In the protocol of drawing 35, if a terminal 23 requires the 1st key list with Command RIS KEYLIST, a host computer 21 will return contents KEYLIST: of the 1st key table with RIS KEYLIST*RESP as a response. In this KEYLIST:, keyword NAME="OS / operating system", "development exchange", a "game", and ... are described corresponding to key number KEY=1, 2 and 3, and ...

[0056] Next, a terminal 23 returns contents KEYLIST[which User A chose with Command RIS KEYLIST / of the 2nd key table corresponding to / host computer / 21 / as a response / the key number 3 in RIS KEYLIST*RESP if the key number 3 of a "game" is sent the 1st keyword]:. In this KEYLIST:, keyword NAME="RPG", "action", "a puzzle/quiz", "a simulation", a "joke", and ... are described corresponding to key number KEY=51, 52, 53, 54 and 55, and ...

[0057] And a terminal 23 requires the list of the software which User A chose with Command RIS LIST and which has delivery and three key numbers 3, 52, and 55 for "action" and the "jesting" key numbers 52 and 55 in a host computer 21 the 2nd keyword. Here, since the host computer 21 has memorized the key number 3 of the 1st already sent keyword, only the key numbers 52 and 55 of the 2nd keyword are sent.

[0058] Drawing 36 shows the 1st and the example of a display of the 2nd keyword. The 1st in drawing 36

and the 2nd keyword differ from them of drawing 35 . If the 1st keyword, such as an image and a game, is displayed and User A chooses the 1st keyword, the 2nd keyword, such as a tool, a text, DOS and WIN, an image, voice, and a game, will be displayed on the screen of the display 24 which User A is looking at next. In the 2nd keyword, an image and the thing which overlaps the 1st keyword like a game are also contained. [0059] User A can also choose the keyword of the gap to display. For example, an image is chosen as the 1st keyword, a game is chosen choosing a tool, DOS, a game, etc. as the 2nd keyword ****, or as the 1st keyword, and DOS, WIN, an image, voice, etc. are chosen as the 2nd keyword. Moreover, User A can also choose two or more keywords as coincidence as the 2nd keyword.

[0060] Since a large number [the keyword for generally choosing the software which he wishes out of the software group 35], it is hard to conclude that these are all displayed at once, and a thing without the need is also displayed. Then, if a role with the keyword important when it displays by making this into the menu of a simple tree structure chosen first is played and selection is mistaken there, software needed will no longer be obtained.

[0061] Then, it enables a user to select a required thing more flexibly out of many keywords by allowing duplication like drawing 36 and displaying a keyword on two hierarchies. In addition, the hierarchy of a keyword may divide and display on more hierarchies but not only two hierarchies.

[0062] Beforehand, the contents of each software of the software group 35 are connected with some keywords, and are stored. For example, in drawing 36 , Software A has a game, an image, and three keywords of DOS, and Software B has an image, a tool, WIN, and four keywords of a game. Therefore, if User A chooses two keywords, an image and a game, the list containing such two software will be sent from a host computer 21, and will be displayed on a screen.

[0063] Drawing 37 shows other examples of the transmitting protocol of the environmental information sent to a host computer 21 from a terminal 23 at the time of host access of step S3. In the protocol of drawing 37 , MODEL in machine information MACHINE: is the information acquired in step S1 at the time of install of terminal software, and expresses the model TOWNS of terminal 23. Moreover, VACANT in PARTINF: is the information on the host computer 21 of step S3 acquired before access (step S2), and expresses the availability of a corresponding partition.

[0064] Drawing 38 shows an example of the protocol of the install initiation in step S17 of the environmental check in step S14, and after that. In the protocol of drawing 38 , a terminal 23 requires delivery and an environmental check of a host computer 21 for number (software code) SOFT=5 of the software which User A chose with Command RIS CHKENV first.

[0065] Directions CHKEXE: for investigating whether a host computer 21 has a file called VBRJP200.DLL in the system directory of a terminal 23 first is sent to a terminal 23 with response RIS CHKENV*RESP. CHKEXE -- : -- **** -- TAG -- = -- " -- VBRJP -- 200 . -- DLL -- " -- a command -- CMD -- = -- " -- ST -- four -- -- D -- : -- \ -- WINDOWS\SYSTEM\VBRJP -- 200 . -- DLL -- " -- a working-level month -- a directory -- assignment -- WORKDIR -- = -- " -- D -- : -- \ -- RIS\KOBUTA -- " -- delivery -- the point -- a directory -- assignment -- SOUKODIR -- = -- " -- D -- : -- \ -- RIS\KOBUTA -- " -- etc. -- describing -- having -- **** .

[0066] A terminal 23 investigates whether file VBRJP200.DLL is in directory WINDOWS\SYSTEM of drive D, makes the result RESULT:, and returns it to a host computer 21. TAG="VBRJP200.DLL=[results-of-an-investigation VAL]-" corresponding to RESULT: is described with " ". This means that there was no file VBRJP200.DLL in a system directory.

[0067] Then, directions ASKCHK: which asks a user whether a host computer 21 may install the selected software is sent to a terminal 23 with response RISCHKENV*RESP. ASKCHK: -- TAG= -- "Q1", question sentence QUERY= which should be displayed" -- in order to perform this software, may I do - install of? "and format ANS: of a reply are described.

[0068] A terminal 23 makes RESULT: the reply which User A inputted, and returns it to a host computer 21

with Command RIS CHKENV. TAG="Q1=[reply result VAL]-" corresponding to RESULT: is described with " ". This means that software may be installed.

[0069] Then, a host computer 21 will send environmental check result JUDGE="O.K." to a terminal 23 with response RIS CHKENV*RESP, if there is no failure on operating environment in others. Except for there being no file VBRJP200.DLL, the operating environment of the software of the software code of No. 5 is ready, and this means that it can install.

[0070] Since the result of an environmental check was set to "O.K.", a terminal 23 requires delivery and install of a host computer 21 for assignment STRPLACE: of classification TYPE="RIS" of the install approach, and a delivery place etc. with Command RIS INSTALL. The working-level month directory WORKDIR and the directory SOUKODIR of a delivery place are described with the software code SOFT of the software installed in STRPLACE:.

[0071] Henceforth, a host computer 21 installs software by the specified approach. In the case of install, as step S18 of [drawing 31](#) explained, a terminal 23 thaws and copies a file with the command sent from a host computer 21, and the serial activity of registering with a system is done at it. Since, as for whether these activities were completed, a terminal 23 returns serially as a response, a host computer 21 can supervise the advance situation of an install activity to the last.

[0072] However, by this approach, since a terminal 23 must perform the communication link with a host computer 21 frequently, communicative effectiveness worsens. Then, the configuration file the host computer 21 described the command of an install activity to be is prepared, and how to pass this to a terminal 23 and to make automatic install perform can be considered.

[0073] [Drawing 39](#) is the flow chart of automatic install which used the configuration file. The install activity of [drawing 39](#) is done after generation of the subdirectory in step S16 of [drawing 30](#).

[0074] A terminal 23 sends Command RIS INSTALL to a host computer 21 with the number of software first (step S21). In response, a host computer 21 starts remote install of the software corresponding to the sent number. Remote install is automatically performed by the exchange between a host computer 21 and a terminal 23 according to the installation script 43 of TETORISU which the host computer 21 created (step S22).

[0075] A host computer 21 transposes @SOUKO@ to SOUKODIR=D:\SOUKO\TET first, and downloads file TET1.LZH to subdirectory D:\SOUKO\TET of a hard disk 25.

[0076] If completion (O.K.) of download is notified from a terminal 23, next, a host computer 21 will transpose @WINDIR@ to D:\WINDOWS, and will download VBRJP200.DLL to directory D:\WINDOWS of a hard disk 25.

[0077] If completion (O.K.) of download is notified from a terminal 23, next, a host computer 21 will transpose @SOUKO@ to SOUKO=D:\SOUKO, and will download the configuration file 44 (SETUP.INF) for user A which described the command of the activity which a terminal 23 should do to directory D:\SOUKO of a hard disk 25. At this time, a host computer 21 rewrites and sends information, such as a model contained in configuration-file SETUP.INF, SOUKO, and WINDIR, to the information for user A with reference to the user A environment file 39 like step S18. File TET1.LZH is thawed in configuration-file SETUP.INF, file TOWNS.DRV is moved and renamed to it, and a series of activities which register file TETRIS.EXE into a system are described.

[0078] When completion (O.K.) of download is notified from a terminal 23, host computers 21 are the directions INSTALL which perform automatic install according to description of configuration-file SETUP.INF next. D:\SOUKO\SETUP.INF is sent to a terminal 23.

[0079] In response, a terminal 23 performs automatic install according to description of configuration-file SETUP.INF. A terminal 23 is Command LOG first. The circuit between host computers 21 is intercepted by OFF. After this, since a communication line 22 is not used, a communication link tariff does not take, either. Next, command LHA X By D:\SOUKO\TET\TET1.LZH, it thaws to four file TETRIS.EXE(s) and

TOWNS.DRV which mentioned file TET1.LZH above, PC98.DRV, and MAC.DRC.

[0080] Next, a terminal 23 is Command MOVE. D:\SOUKOTET\TOWNS.DRV By D:\WINDOWS\FONT.DRV, file TOWNS.DRV is moved to D:\WINDOWS from directory D:\SOUKOTET, and the file name is changed into FONT.DRV. Next, command ICON By TETRIS.EXE, file TETRIS.EXE is iconified and it registers into a terminal 23. And a power source is turned OFF with Command POFF, and an install activity is ended.

[0081] In automatic install using a configuration file, although a host computer 21 cannot carry out the last check of activity termination, the utilization ratio of a communication line 22 can be raised by reducing the count of a communication link. Moreover, install conditions, such as a model, are solved before the purchase of software, and, after that, install is performed automatically.

[0082] Drawing 40 shows an example of the configuration file of other formats. In the configuration file of drawing 40, [DstDirs] is a list of storing places of a file, and [Files] is a list of files which should be stored. Here, storing in directory D:\RIS\KOBUTA execution file Soft2.Exe of software which a user wishes, changing into FONT.DRV the name of TOWNS.DRV which is a driver for TOWNS, and storing in directory D:\WINDOWS\SYSTEM is described.

[0083] Although described as 1=SOUKODIR and 2=WINDIR by the generate time of a configuration file, when [DstDirs] acquires the information on SOUKODIR with an actual host computer 21, and WINDIR, it is rewritten with 1=D:\RIS\KOBUTA and 2=D:\WINDOWS\SYSTEM by it. Thus, the setups in a configuration file can be changed dynamically or alternatively.

[0084] As explained above, it becomes possible to install automatically in the terminal unit the software which a user wishes through a communication line from prior ***** and a delivery center. It also becomes possible using this to sell software to a user through a communication line.

[0085] At this time, since the delivered software is stored in the directory of dedication, its maintainability is good, and a user does not need to input the character string which specifies the directory of the address for delivery etc.

[0086] A user can select efficiently the software which he wishes on the screen of a terminal unit, and can also choose the approach of the install. Moreover, a user does not need to check the operating environment of software and an environmental check is performed automatically.

[0087] Furthermore, use of a communication line can be lessened as much as possible at the time of install of software.

[0088]

[Problem(s) to be Solved by the Invention] However, there are the following problems in an above-mentioned remote install system.

[0089] There is a thing of various classes in the software which is circulating by current and personal computer communications. For example, what is called freeware is distributed fundamentally for nothing, and if a predetermined price is remitted, the functional limit is to be canceled once what is called shareware is sent with a functional limit for nothing. Moreover, what is sold as goods is fundamentally sent in exchange for price.

[0090] When offering service in which a delivery center sells software to a user using the function of remote install, corresponding to such various selling gestalten, the device in which it guarantees receiving price certainly is needed. How in the case of shareware, the reception of price and discharge of a functional limit are processed especially poses a problem.

[0091] This invention aims at offering the software price sanction system which sanctions the price of the delivered software automatically, and its approach in the service which sells software through a communication line.

[0092]

[Means for Solving the Problem] Drawing 1 is the principle Fig. of the software price sanction system of

this invention. The software price sanction system of drawing 1 is formed in the information processing system which delivers software through a communication network to a user's terminal unit, and is equipped with the software information storage means 51, the price information storage means 52, the processing means 53, and the accounting means 54.

[0093] The software information storage means 51 memorizes the software information containing the body file and management information of registration software. The price information storage means 52 memorizes the price information about the selling gestalt and price of the above-mentioned registration software.

[0094] When the delivery demand of the above-mentioned registration software is received from the 1st user, the processing means 53 determines the price of the registration software based on the above-mentioned price information memorized by the price information storage means 52, and performs delivery processing according to the above-mentioned selling gestalt.

[0095] The selling gestalten and prices of registration software, such as freeware and shareware, are described, and the processing means 53 can determine the price of the software which received the delivery demand with reference to this price information, and can perform suitable delivery processing in the above-mentioned price information. For example, when registration software is freeware, it installs in the 1st user's terminal for nothing, and in the case of goods, it installs in exchange for the specified price. Moreover, in the case of shareware, processing required for a predetermined price and discharge of the functional limit to exchange is performed. In this way, the price of software is sanctioned automatically.

[0096] The processing means 53 receives beforehand the above-mentioned software information and price information from the terminal of the 2nd user who registers software through a network, and registers them into the software information storage means 51 and the price information storage means 52, respectively.

[0097] Moreover, the accounting means 54 charges automatically the price which the processing means 53 determined to the 1st user of the above. Thereby, it enables a delivery center or the 2nd user to receive the price of charged software certainly.

[0098] For example, the software information storage means 51 and the price information storage means 52 of drawing 1 correspond to the storage which accompanies the host computer 63 in drawing 2 of an operation gestalt, and the processing means 53 and the accounting means 54 correspond to CPU (central processing unit) of a host computer 63.

[0099]
[Embodiment of the Invention] Hereafter, the operation gestalt of this invention is explained to a detail, referring to a drawing. Drawing 2 is the remote install structure-of-a-system Fig. of an operation gestalt. In the system of drawing 2, the terminal 61-1 of N individual, ..., 61-N have combined with the host computer 63 through a communication network 62. A terminal 61-1, ..., 61-N are the terminal units of each user set as the object of service, and a host computer 63 is in a delivery center side. Although a host computer 63 is visible to one set if it sees from a network 62, distributed processing by two or more computers is also possible in fact. First, how to register software (contents) into a host computer 63 through a network 62 is explained. Although the author of software needs to upload information required for registration with the created body file, he standardizes the information to upload at this time. It standardizes, and TAG which shows the contents is attached to the information classified into some, and it is made to send to it as a concrete approach of uploading software information. By standardizing information, it becomes automatable [registration and a check activity]. Moreover, by dividing and uploading information, only a part required at the time of discovery of a bug can be resent, and the time amount which upload for the second time takes can be shortened. Upload information is roughly classified and becomes the following five kinds. [1] CFG file (configuration file) : the following are included by the basic intelligence (indispensable information) for registering software into a system.

O The previous version of the information contents name contents for registering with a database, the

present version (it is not indispensable)

Selling gestalt (freeware / shareware / commercial item)

The existence of information pretreatment of the pretreatment and after treatment at the time of approach O install of the compression approach install of the information contents body of this processing of OS retrieval keyword O install in which the hardware actuation in which price actuation is possible, the existence of approach after treatment, and an approach [2] explanation file: Explanation which wrote the outline of software (indispensable information)

[3] Body file : what compressed the file required for activation and was summarized to one (indispensable information)

[4] CHK file (check file) : since the environmental condition to which software operates is investigated and remote install is embellished, only software with an environmental dependency is needed by the file written in special language. Although what is often used is offered by the system, a user creates a special thing as an option. The following information is included in a CHK file, for example.

[0100] CHKEXE: It is created when version information of executive program VERSION:DLL for the environmental check performed as an option (dynamic link library) needs to be checked as an option.

[0101] Information like the check script 40 of [drawing 30](#) can also be described to a CHK file.

[5] Install related file : it is what wrote the actual procedure of remote install, and the following information is included.

[0102] INSTALL: The procedure of actual install (unnecessary, when a procedure is standardized)

The standardized install processing can be as follows and a great portion of software can be installed within the limits of this.

[0103] 1. If specified, it will pretreat.

2. Download the compressed body to the work-piece field of a terminal.

3. Develop a body and copy a file to a suitable directory.

[0104] 4. A file attribute will be changed if there is assignment.

5. If there is assignment, a menu and icon registration will be performed.

6. After treatment will be performed if there is assignment.

[0105] ICON: When information FIRST:INSTALL for registering software is standardized to what is called a menu and manager, it is pretreatment performed before standard install processing, for example, point out the backup process of an old edition etc. to it in version up processing.

[0106] When LAST:INSTALL is standardized, it is the after treatment performed after standard install, for example, rewriting processing of environment variable file CONFIG.SYS etc. is pointed out.

Next, the example of a file is explained, referring to from [drawing 3](#) to [drawing 10](#). [Drawing 3](#) to [drawing 6](#) shows the example of a CFG file, and [drawing 7](#), and 8, 9 and 10 show the example of an explanation file, a body file, a CHK file, and an install related file to it, respectively.

[0107] In [drawing 3](#) and the CFG file of 4, 5, and 6, it expresses that the line which starts in; is a comment sentence. [name] of [drawing 3](#) expresses the name for software retrieval, [version] expresses the version and [oldversion] expresses the earlier version. Moreover, [type] expresses a selling gestalt and [price] expresses a price. Here, [type] serves as SHARE and it is shown that it is shareware. [machine] expresses the hardware which can operate and 4 models, TOWNS, FMR, PC98, and PC-AT, are mentioned here.

[0108] In [drawing 4](#), [os] expresses OS which can operate and only WIN31J are mentioned here. [key] expresses the keyword for retrieval and "Chinese" is describing it as a keyword here. [compression] expresses the compressed format of a body and is describing it as the LHA method here. Moreover, [instype] expresses the approach of possible remote install. "RIS icon.def" expresses how to perform the usual remote install and register into icon file icon.def, "INSTALLER icon2.def" expresses how to carry out icon registration of the installer (INSTALL.EXE) itself, and "DLOAD" expresses how to download. here -- any of these approaches -- although -- it is possible.

[0109] In drawing 5 , [souko] expresses the recommendation directory which stores software at the time of install, and two directories, P1 and P2, are describing it here.

[0110] [source] expresses the information on the compressed body file (archive file), and the information on two archive files, S1 and S2, is describing it here. The file name of S1 is NEWWP.LZH and, for the storing directory after expansion (after defrosting), the mode at the time of @P1@ and expansion is [the minimum need capacity after 800000 bytes and expansion of M and file capacity] 2 million bytes. Moreover, the file name of S2 is NEWWPLIB.LZH and, for the mode at the time of @P2@ and expansion, F and file capacity are [the storing directory after expansion / the object file names of 13000 bytes and a duplication check of the minimum need capacity after 7000 bytes and expansion] GSWDLL.DLLs.

[0111] Here, if Mode M is specified, an expansion place directory can be changed by the terminal side, and if Mode F is specified, it cannot change. Moreover, GSWDLL.DLL is one of the files by which compression storing is carried out in archive file NEWWPLIB.LZH, and it is described in order to check for no duplication of a file name on a terminal.

[0112] [destination] expresses a special expansion place directory and @WINSYSTEM@ is describing it as D1 here. Moreover, [copy] expresses information, such as a file which should make it move at the time of install, and moving file GSWDLL.DLL contained in the archive file specified by S2 as C1 by the same file name as the directory specified by D1 is describing here. N of the column of Attr expresses that an attribute is eternal also even for after migration.

[0113] In drawing 6 , [first] expresses the information on pretreatment and a command to execute before download of an archive file in the case of install describes it. [last] expresses the information on after treatment and a command to execute after icon registration command activation in the case of install describes it.

[0114] The explanation file of drawing 7 is the new word processor described by the CFG file from drawing 3 to drawing 6 . - It is the information for explaining to a user the outline of software called A2, and the body file of drawing 8 shows the archive file. Archive file NEWWP.LZH contains three file WP.EXE(s) and NEWWP.ICO(s), and README.DOC, and it turns out that archive file NEWWPLIB.LZH contains only one file GSWDLL.DLL.

[0115] The CHK file of drawing 9 corresponds to the check script of the environmental check at the time of install. Here, if it is with the check of whether a file called WP.EXE is in a user's terminal, the check of being the version of the newest [it] and the processing according to each check result are described.

[0116] Drawing 10 shows the ICON file which is a kind of an install related file. In the ICON file of drawing 10 , it is a new word processor. - Information, such as a file for the execution file of A2, a current working directory, and icon registration, is described. This ICON file is used in the case of the system registration at the time of remote install.

[0117] It is a new word processor by uploading such a CFG file, an explanation file, a body file, a CHK file, and an install related file from a terminal to a host computer 63. - A2 is registered automatically. The registered software is delivered by the terminal by onerous or onerous according to the demand from a user.

[0118] By the way, when selling software by the remote install system, according to the class of software, the purchase hysteresis of a user's past, etc., a special procedure for accounting and modification of a price may be needed. Then, in order to perform accounting according to the class of software to deliver, the file which defined the procedure about accounting further is uploaded at the time of software registration.

[0119] Drawing 11 shows the example of the file group created on terminal 61-i (i= 1, 2, ..., N). In addition to the CFG file 72, the explanation file 73, the body file 74, the CHK file 75, and the install related file 76, in drawing 11 , the definition file 77 and the rewriting file 78 are created within memory 71. And these file groups upload to a host computer 63.

[0120] Here, the case where the author of shareware software registers it into a host computer 63 is considered. In the case of shareware, the author describes the remittance procedure of shareware etc. to the

definition file 77. The approach of update of a software name, a software code, a remittance place, the remittance amount of money, the type of remittance, and a user system is described by the definition file 77. [0121] Drawing 12 shows the example of the definition file 77 for shareware. In definition file AAAS.CFG of drawing 12, [name] expresses the name of shareware "an AAA scheduler" and remittance procedure is described by [type]. Here, FJOKI is describing as ID of the remittance place to the software code 555. AAA expresses the label of assignment **** and means the "AAA scheduler" here. [price] expresses a price, [machine] and [os] express the model and OS which can operate, respectively, and [key] expresses the keyword for retrieval.

[0122] The type of remittance is described by [instype], and if a remittance request is carried out, the processing of which a functional limit is canceled on that spot is describing here. Moreover, update processing of a user file is described by [last] as after treatment of install. Here, the processing which rewrites a user's file according to the contents described by file INI.DEF is describing using Command CHGINI.

[0123] In addition, the difference in the remittance procedure of the usual software and shareware is identified with the section of [type]. The CHK file 75 is used also as one of the options files about remittance procedure, and the contents which a host computer 63 exchanges with a user interactively at the time of install are described. Drawing 13 shows the example of such a CHK file 75. In CHK file AAAS.CHK of drawing 13, it investigates whether file AAA.INI is in a user system, and the processing which checks the storing location to a user is described. AAA.INI is a file for setting up the function of shareware, and unless it writes right information in the section of the [UserInfo], shareware does not operate completely.

[0124] The rewriting file 78 is an options file which defines an approach to rewrite a user system. Drawing 14 shows the example of the rewriting file 78. The processing which rewrites the value of Licence of the section of [UserInfo] in file AAA.INI of a user system to AA123AAA is described by rewriting file INI.DEF of drawing 14. Licence=AA123AAA plays the role of the code code of which a functional limit of shareware is canceled.

[0125] Drawing 15 shows the example of the upload processing of a file group shown in drawing 11. In drawing 15, the author of shareware first uploads the CFG file 72 (AAA.CFG), the explanation file 73 (AAA.TXT), the install related file 76 (here icon file ICON.DEF), and the body file 74 (AAA.LZH) from his own terminal as a body catalogued file. A host computer 63 registers such upload information into the contents database 81. Here, the software code of the uploaded software "an AAA scheduler", a name, the type (TYPE), the body file name, the explanation file name, the icon file name, etc. are registered as management information. SHARE of the column of a type expresses that the class of software is shareware. The contents database 81 is formed for example, in a host computer 63 or an external disk unit.

[0126] Next, the author uploads the definition file 77 (AAAS.CFG), the CHK file 75 (AAAS.CHK), and the rewriting file 78 (INI.DEF) as a remittance procedure file. Thereby, the software code of the remittance procedure file AAAS, a name, a type, a CHK file name, the file name for after treatment, etc. are registered into the contents database 81. As a file for after treatment, file name INI.DEF of the rewriting file 78 is describing here. Moreover, SOKIN#RIS of the column of a type expresses that the class of software is the remittance procedure file of shareware, and supports the information on [instype] of AAAS.CFG.

[0127] Next, the remittance procedure of the shareware using the uploaded file group is explained, referring to from drawing 16 to drawing 20. In addition to the procedure of remote install of point **, in shareware procedure, a remittance flag is formed on a protocol. And it enables it to choose shareware procedure by setting this flag from a terminal and requiring retrieval of software. Moreover, the menu which carries out ON/OFF of the remittance flag is displayed on the screen of a terminal.

[0128] Drawing 16, and 17, 18 and 19 show the remittance procedure of shareware "an AAA scheduler." However, shareware "an AAA scheduler" shall be installed in a user system with a functional limit before

remittance procedure. When a user tries to purchase this software, the command/a response like illustration between a host computer 63 and a terminal are exchanged.

[0129] In drawing 16 , a terminal transmits a user environment to a host computer 63 first (step S31), and a host computer 63 will return a response, if this is received (step S32). Next, if a user demands the keyword list for software retrieval (step S33), a host computer 63 will return a keyword list (step S34).

[0130] At this time, as shown in drawing 17 , the menu 82 of option procedure is displayed on the screen of a terminal, and a user specifies "remittance" from that inside and chooses a keyword from keyword lists. Thereby, the remittance flag SOKIN is set (turned on) and the directions which start retrieval of shareware are sent to a host computer 63 (step S35). A host computer 63 returns the name and the software code of a remittance procedure file (remittance software) of the software with which the contents database 81 is searched with the specified keyword, and a type starts in SOKIN (step S36). Here, names, such as an "AAA scheduler" and a "BBB scheduler", and the software code 4000 of remittance software, and 4001 grades are returned.

[0131] Carrying out remittance soft chisel listing of the terminal, a user specifies the specific remittance software of the inside by which listing was carried out (step S37). Here, the software code 4000 is specified. A host computer 63 performs a negotiation a terminal side according to the conditions of the remittance software of the software code 4000 (step S38). Here, it directs to investigate the location of AAA.INI in a user system using a command ST 4 to a terminal first. In response, a terminal investigates the location of AAA.INI and the location notifies E:\AAA to a host computer 63 (step S39). A command called ST4 shall be beforehand provided in the terminal side.

[0132] Next, in drawing 18 , a host computer 63 displays a dialog BOX83 on the screen of a terminal, and it checks to a user whether the location of AAA.INI is good in E:\AAA (step S40). If the directory displayed on the dialog BOX83 is right, a user will return the directory as it is (step S41), and a host computer 63 will decide with E:\AAA\AAA.INI the directory path of the file which should rewrite a user system. If the displayed directory is not right, a user inputs a right directory name. For example, if G:\GGG is inputted, a terminal will return directory path G:\GGG\AAA.INI to a host computer 63. Since the storing location of AAA.INI was decided, a host computer 63 notifies to a terminal that shareware remittance is possible (step S42).

[0133] Next, in drawing 19 , a user demands discharge of a functional limit (step S43). After a host computer 63 charges price directly to the user's account etc. with reference to [instype] of the definition file of drawing 12 in response, rewriting file INI.DEF the rewriting procedure of AAA.INI is described to be is sent. Furthermore, it directs to rewrite the command CHGINI of after treatment according to delivery and the procedure of INI.DEF with reference to [last] of a definition file to a terminal.

[0134] In response, with reference to downloaded INI.DEF, a terminal rewrites AAA.INI, as shown in drawing 20 . In drawing 20 , it turns out that AA123AAA is written in Licence of [UserInfo] after rewriting. Thereby, a functional limit of shareware "an AAA scheduler" is canceled and it comes to operate completely on a user system.

[0135] Then, a host computer 63 deletes INI.DEF from a terminal, and ends processing. In the definition file of drawing 12 , since the information that limit discharge was performed on that spot was describing into the section of [instype], the functional limit of shareware was canceled to pulling down and coincidence of price. However, a host computer 63 can also consider as the configuration which performs only issue of the electronic mail to price pulling down and a shareware registrant like a common personal computer communications center.

[0136] Drawing 21 shows the example of the definition file 77 in the case of supporting only price pulling down and issue of an electronic mail. In definition file BBBS.CFG of the "BBB scheduler" of drawing 21 , it is the same as that of drawing 12 about the section from [name] to [key]. Only remittance processing is describing at [instype] and discharge of a functional limit on the spot is not performed. Moreover, the author

of a "BBB scheduler" does not need to prepare drawing 13, a CHK file like 14, and a rewriting file in this case. If definition file BBBS.CFG uploads to a host computer 63, it will be recorded on the column of the type of the contents database 81 as SOKIN#MAIL.

[0137] Drawing 22, and 23 and 24 show the price pulling-down procedure of software "a BBB scheduler." However, shareware "a BBB scheduler" shall be installed in a user system with a functional limit before this procedure. When a user tries to purchase this software, the command/a response like illustration between a host computer 63 and a terminal are exchanged.

[0138] In drawing 22, a terminal transmits a user environment to a host computer 63 first (step S51), and a host computer 63 will return a response, if this is received (step S52). Next, if a user demands the keyword list for software retrieval (step S53), a host computer 63 will return a keyword list (step S54).

[0139] At this time, as shown in drawing 23, the menu 82 of option procedure is displayed on the screen of a terminal, and a user specifies "remittance" from that inside and chooses a keyword from keyword lists. Thereby, the remittance flag SOKIN is set and the directions which start retrieval of shareware are sent to a host computer 63 (step S55). A host computer 63 returns the name and the software code of remittance software of the software with which the contents database 81 is searched with the specified keyword, and a type starts in SOKIN (step S56). Remittance soft chisel listing of the terminal is carried out, and a user specifies the remittance software of the software code 4001 from from, while listing was carried out (step S57).

[0140] Next, in drawing 24, it checks to a user whether a host computer 63 may display a message 84 on the screen of a terminal, and may be charged (step S58). At this time, a host computer 63 changes the value of Flag SOKIN into "0x08" only showing e-mail issue with reference to [instype] of the definition file of drawing 21, and returns it to a terminal. If a user may be charged, when not purchasing O.K., he chooses NG. Here, O.K. is chosen and the request of issue of e-mail to a registrant is sent to a host computer 63 from a terminal (step S59). In response, a host computer 63 charges price directly to the user's account etc., and sends the electronic mail which tells the registrant of a "BBB scheduler" about price pulling down. As the receiver's address of e-mail, the remittance place FJOKI of the section of [type] of a definition file is used.

[0141] The registrant who received the electronic mail from the host computer 63 connects the approach of functional limit discharge to the purchaser of software with means, such as an electronic mail. Thereby, a purchaser can use all the functions of a "BBB scheduler" now.

[0142] By the way, it may be to change a setup of the selling price to the user by whether the earlier version of software with a user or related software is already purchased. In this case, the registrant of software describes two or more price and its selling condition into the section of [price] of the definition file 77. For example, the changing condition of a price can be described using the software code of purchased software, user ID (UID), and Machine ID (MID). UID is an identifier which specifies a user as a meaning here, and MID is an identifier which specifies the terminal of an install place as a meaning. The approach of two or more price setup using such a definition file 77 is applicable not only like the remittance procedure of shareware but a price setup of the ISV (independant software vender) software sold as usual goods.

[0143] Drawing 25 shows the example of the definition file 77 which set up two or more prices. It expresses that the section of [type] of the "CCC scheduler" of drawing 25 is ISV software. Moreover, in the section of [price], the conditions for choosing the selling price are describing it as three prices, 0 yen, 2000 yen, and 5000 yen, from those inside. The selection condition of the selling price is as follows.

(a) It is no charge when the user and terminal which have been accessed have already bought the software XSOFT of the software code 111. XSOFT may be the software used as the predecessor for example, of a CCC scheduler, and may be the earlier version of a CCC scheduler.

(b) It is 2000 yen when the accessed user or terminal buys the software YSOFT of the software codes 123 or 124. YSOFT is the related software of a CCC scheduler.

It is 5000 yen when it does not correspond to (c), (a), or (b). Moreover, the database of a host computer 63 is constituted as shown in drawing 26. In drawing 26, the customer database 85 memorizes the management information about the user and terminal of a remote install system, and the purchase database 86 memorizes the purchase hysteresis about the user who purchased software in the past. Moreover, the storing place of the price file 87 which described the selection condition of a price is recorded on the column of PRICE of the CCC scheduler of the contents database 81. These databases 85, 86, and 81 are formed in a host computer 63 or an external disk unit, and the price file 87 is also stored in the disk unit. A host computer 63 reads the price file 87 of a CCC scheduler with reference to the contents database 81, and determines a price according to the description.

[0144] Drawing 27 is the flow chart of the price determination processing based on the price file 87. if processing is started in drawing 27, with reference to the customer database 85 and the purchase database 86, a user's accessed UID and MID of the terminal should put a host computer 63 together first -- it comes out and investigates whether the software of the software code 111 may be purchased (step S61). If there are such historical data, processing will be ended using a price as 0 yen (step S62). If there are such no historical data next, it will investigate whether the record of purchase of the software of the software codes 123 or 124 is in the purchase hysteresis containing a user's accessed UID or MID of the terminal (step S63). If there are such historical data, processing will be ended using a price as 2000 yen (step S64). And if there are such no historical data, processing will be ended, using a price as 5000 yen (step S65).

[0145] Now, UID of the user who wishes the purchase of a CCC scheduler is set to FJOTA, and the MID is set to 0005. Since it is recorded on the purchase database 86 of drawing 26 that the user purchased the software of the software code 111 using the terminal, the judgment result of step S61 serves as YES, and a CCC scheduler is delivered for nothing.

[0146] When the user who has purchased neither of the software of the software codes 111, 123, and 124 accesses from the terminal which has purchased those neither, a price changes to 5000 yen of the highest frame. Moreover, if one of the two is not in agreement with the identifier used for access among the UID and MID even if there is purchase hysteresis of the software code 111, software serves as a charge.

[0147] It may not pass over the definition file of drawing 25 to an example of a price setup, but a registrant may set up the selection condition of other arbitration combining a software code, and UID and MID. For example, to specific UID or specific MID, it can also set up so that it may sell at a price specially. Thus, various price setup of software is attained by describing and uploading the setups of a price to the definition file 77.

[0148] Moreover, the software for sale is not ISV software, and when it is shareware, based on the price file, a price is determined in the case of price pulling down in drawing 19 or drawing 24.

[0149]

[Effect of the Invention] According to this invention, in the service which sells software through a communication line, the price of various software can be determined automatically and accounting can be performed. After charging price directly especially in the case of shareware, it becomes possible to cancel a functional limit.

[0150] Moreover, the registrant of software can specify the sanction approach of price and can register the information beforehand. For example, distinction of freeware, shareware, and charged software, the setups of a price, etc. are specified.

[Translation done.]

Cited Reference 1

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-101986

(43) 公開日 平成9年(1997)4月15日

(51) Int.Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 17/60 9/445			G 0 6 F 15/21 9/06	Z 4 2 0 J

審査請求 未請求 請求項の数14 O L (全 38 頁)

(21) 出願番号 特願平7-258507

(22) 出願日 平成7年(1995)10月5日

(71) 出願人 000005223
富士通株式会社
神奈川県川崎市中原区上小田中4丁目1番1号

(72) 発明者 沖 宏志
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(72) 発明者 鎌田 紳二
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(74) 代理人 弁理士 大曾 義之 (外1名)

最終頁に続く

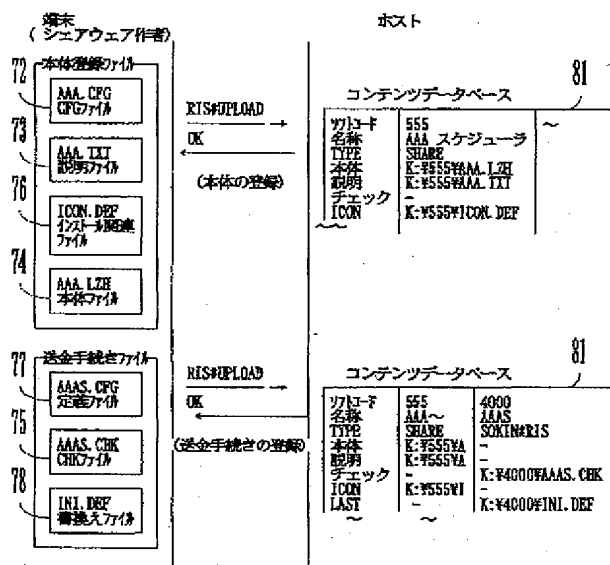
(54) 【発明の名称】 ソフトウェア代金決裁システムおよび方法

(57) 【要約】

【課題】 通信回線を介してソフトウェアを販売するサービスにおいて、配送したソフトウェアの代金を自動的に決裁することが課題である。

【解決手段】 ソフトウェアの作者は、本体登録ファイルと送金手続きファイルとを、通信ネットワークを介してホスト計算機にアップロードする。ホスト計算機は、これらの情報をコンテンツデータベース81に登録し、ユーザから配送要求を受け取ると、ソフトウェアのタイプに応じて課金・配送を行う。例えばシェアウェアの場合は、送金手続きファイルの情報に従って、代金引き落としと機能制限の解除とが同時に行われる。また、ホスト計算機は、価格決定のアルゴリズムを記述した価格ファイルを記憶することにより、多様な価格設定を実現することができる。

アップロードを示す図



【特許請求の範囲】

【請求項1】 ユーザの端末装置に通信ネットワークを介してソフトウェアを配送する情報処理システムにおいて、

登録ソフトウェアの本体ファイルと管理情報とを含むソフトウェア情報を記憶するソフトウェア情報記憶手段と、

該登録ソフトウェアの販売形態および価格に関する価格情報を記憶する価格情報記憶手段と、

第1のユーザから前記登録ソフトウェアの配送要求を受け取った時、前記価格情報記憶手段に記憶された前記価格情報に基づいて該登録ソフトウェアの価格を決定し、前記販売形態に応じた配送処理を行う処理手段とを備えることを特徴とするソフトウェア代金決済システム。

【請求項2】 前記処理手段が決定した代金を前記第1のユーザに対して自動的に課金する課金手段をさらに備えることを特徴とする請求項1記載のソフトウェア代金決済システム。

【請求項3】 前記処理手段は、前記ソフトウェア情報および価格情報を第2のユーザの端末装置から受け取り、前記ソフトウェア情報記憶手段および価格情報記憶手段に登録することを特徴とする請求項1記載のソフトウェア代金決済システム。

【請求項4】 前記登録ソフトウェアがシェアウェアの場合、前記価格情報記憶手段は、シェアウェアの機能制限の解除手続きを含む前記価格情報を記憶し、前記処理手段は、代金の決済時に該解除手続きに基づいて必要な処理を行うことを特徴とする請求項1記載のソフトウェア代金決済システム。

【請求項5】 前記処理手段は、前記第1のユーザの端末からシェアウェア手続き用のフラグを受け取り、該フラグが立っていれば、前記シェアウェアの機能制限の解除処理を行うことを特徴とする請求項4記載のソフトウェア代金決済システム。

【請求項6】 前記処理手段は、前記第1のユーザの端末の画面上にシェアウェア手続きを選択するメニューを表示させ、該第1のユーザが該シェアウェア手続きを選択すると、該第1のユーザの端末から送られるシェアウェア手続き用のフラグが立つことを特徴とする請求項5記載のソフトウェア代金決済システム。

【請求項7】 前記処理手段は、前記解除処理において、前記シェアウェアの制限解除に必要なファイルを前記第1のユーザの端末に送り、該第1のユーザの端末に対して機能制限の解除を指示することを特徴とする請求項5記載のソフトウェア代金決済システム。

【請求項8】 前記処理手段は、前記解除処理において、前記第1のユーザに代金を課金したことを通知する電子メールを、前記シェアウェアの登録者に対して発行することを特徴とする請求項5記載のソフトウェア代金決済システム。

【請求項9】 ユーザの購入したソフトウェアのソフトウェア識別子を含む購入履歴情報を記憶する購入履歴記憶手段をさらに備え、前記処理手段は、前記第1のユーザのユーザ識別子と該第1のユーザの端末のマシン識別子とに応じて、前記購入履歴情報に含まれるソフトウェア識別子を参照し、前記価格情報に従って前記登録ソフトウェアの価格を決定することを特徴とする請求項1記載のソフトウェア代金決済システム。

【請求項10】 ユーザの購入したソフトウェアのソフトウェア識別子を含む購入履歴情報を記憶する購入履歴記憶手段をさらに備え、前記処理手段は、前記第1のユーザのユーザ識別子に応じて前記購入履歴情報に含まれるソフトウェア識別子を参照し、前記価格情報に従って前記登録ソフトウェアの価格を決定することを特徴とする請求項1記載のソフトウェア代金決済システム。

【請求項11】 ユーザの購入したソフトウェアのソフトウェア識別子を含む購入履歴情報を記憶する購入履歴記憶手段をさらに備え、前記処理手段は、前記第1のユーザの端末のマシン識別子に応じて前記購入履歴情報に含まれるソフトウェア識別子を参照し、前記価格情報に従って前記登録ソフトウェアの価格を決定することを特徴とする請求項1記載のソフトウェア代金決済システム。

【請求項12】 ユーザのシステムに通信ネットワークを介してソフトウェアを配送するソフトウェア配送センターに、該通信ネットワークを介してソフトウェアを登録する情報処理システムにおいて、

登録ソフトウェアの本体ファイルと管理情報とを含むソフトウェア情報と、該登録ソフトウェアの販売形態および価格に関する価格情報とを記憶する記憶手段と、

該ソフトウェア情報および価格情報を、前記通信ネットワークを介して前記ソフトウェア配送センターにアップロードするアップロード手段とを備えることを特徴とする端末装置。

【請求項13】 ユーザの端末装置に通信ネットワークを介してソフトウェアを配送する情報処理システムにおける記憶媒体であって、

登録ソフトウェアの本体ファイルと管理情報とを含むソフトウェア情報と、該登録ソフトウェアの販売形態および価格に関する価格情報とを、第1のユーザから受け取る受信手段と、

第2のユーザから前記登録ソフトウェアの配送要求を受け取った時、前記価格情報に基づいて該登録ソフトウェアの価格を決定し、前記販売形態に応じた配送処理を行う処理手段とを備えることを特徴とする記憶媒体。

【請求項14】 ユーザの端末装置に通信ネットワークを介してソフトウェアを配送する方法において、

登録ソフトウェアの本体ファイルと管理情報とを含むソフトウェア情報と、該登録ソフトウェアの販売形態および価格に関する価格情報とを、第1のユーザから受け取

り、
第2のユーザから前記登録ソフトウェアの配送要求を受け取った時に、前記価格情報に基づいて該登録ソフトウェアの価格を決定し、前記販売形態に応じた配送処理を行うことを特徴とするソフトウェア代金決裁方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、通信回線を介してソフトウェアをユーザの端末装置にインストールするサービスに係り、ソフトウェアの代金を決裁するソフトウェア代金決裁システムとその方法に関する。

【0002】

【従来の技術】近年のパソコン通信等の発達に伴い、通信回線を介してソフトウェアを入手したいというユーザの希望が生じているが、このようなソフトウェアの配送を自動的に行う本格的なシステムは実現していない。ソフトウェアの自動配送を行うには、まずユーザが希望するソフトウェアを自宅の端末装置から配送センターに知らせ、ユーザから通知を受けた配送センターは通信回線を介して要求されたソフトウェアをユーザの端末装置に送信するという形式が考えられる。

【0003】このようなソフトウェアの自動配送を実現するために、先願の「リモートインストールシステムおよび方法」（特願平7-1797）が考案された。以下、図面を参照しながら先願の技術について説明する。

【0004】図28は、先願のリモートインストールシステムの構成図である。図28のリモートインストールシステムは、ソフトウェアの配送センターにあるホスト計算機21、ユーザの端末23、およびこれらを結ぶ通信回線22から成る。端末23は、例えば配送センターから遠く離れたユーザの自宅にあるパーソナルコンピュータであり、通信回線22は例えばパソコン通信のネットワーク用の回線であるものとする。

【0005】ホスト計算機21は、図示されない格納領域内に配送可能な複数のソフトウェアを含むソフトウェア群35と、そのソフトウェア群35から特定のソフトウェアを選択するときに用いるキーワードのリストを保持する第1キーテーブル32および第2キーテーブル33を格納している。

【0006】ユーザが端末23からホスト計算機21にキーワードのリストを要求すると、ホスト計算機21は第1キーテーブル32、第2キーテーブル33を順次送信して、それらに含まれるキーワードを端末23の表示装置24の画面に表示させる。ユーザは表示されたキーワードから希望するソフトウェアに対応するものを選び、ホスト計算機21に通知する。

【0007】ホスト計算機21は通知されたキーワードに該当するいくつかのソフトウェアの名称を含むメニューを表示装置24の画面に表示させ、ユーザはの中から希望するソフトウェアを選んで、ホスト計算機21に

通知する。そして、ホスト計算機21はソフトウェア群35からユーザの選んだソフトウェアのコンテンツ（ファイル）を取り出し、端末23のハードディスク25に設定された配送（宅配、またはディストリビューション）用のディレクトリD：¥SOUKOに格納する。

【0008】このとき、宅配されたソフトウェアを起動するためのアイコン37が自動的に登録され、表示装置24の画面上でディレクトリD：¥SOUKOに対応して設けられた倉庫ウィンドウ38内に表示される。例えば端末23がWINDOWSを搭載している場合は、宅配されたソフトウェアはWINDOWSのプログラムマネージャに登録される。以後、ユーザはアイコン37をマウス等の入力装置を用いてクリックするだけで、宅配されたソフトウェアを使用することができる。

【0009】端末23が装備するハードウェアやソフトウェアツール等の動作環境を記述した環境ファイル31は端末23内で生成され、ホスト計算機21のアクセス時にホスト計算機21に送られる。ホスト計算機21は受け取った環境ファイル31を各ユーザ毎に保持し、ユーザが選んだソフトウェアが動作可能かどうかのチェックに用いる。

【0010】また、ホスト計算機21はユーザ毎に個別の設定ファイル34を用意し、ユーザが選んだソフトウェアの動作に必要なファイル一式とその格納場所をこれに記述する。そして、それらのファイル一式とともに設定ファイル34を端末23に一括して送り、端末23にソフトウェアの自動インストールを行わせることができる。

【0011】次に、図29から図31までを参照しながら、図28のリモートインストールシステムによるソフトウェアの宅配の動作フローを説明する。図29において、ユーザAの端末23は通信用の端末ソフトのインストール時に動作環境の情報を取得し、取得した情報を設定した環境ファイル38を作成する（ステップS1）。このとき、ユーザの端末の機種や宅配に使用する格納場所SOUKO等の取得に時間のかかる情報、あるいは、場合によってユーザに問い合わせなければならない情報を取得する。

【0012】格納場所SOUKOの決定にあたっては、端末23はハードディスク25に所定の容量以上の空き領域があるかどうかを調べ、空き領域があればそのルートに宅配用のディレクトリを作成する。このときディレクトリ名等は端末23が自動的に生成し、ユーザAはそれを確認する作業のみを行う。したがって、ユーザAはディレクトリ名等を入力する必要がない。

【0013】ここでは、ユーザAの機種はTOWNSであり、SOUKOのディレクトリはD：¥SOUKO（ドライブDのディレクトリSOUKO）であることが環境ファイル38に書き込まれる。ユーザAは、必要があればD：¥SOUKOを他のディレクトリに変更する

こともできる。

【0014】既に設定されているパーティションに所定の容量の空き領域がなければ、別のパーティションの空き領域が一番大きい場所を探して、そこに宅配用のディレクトリを作成する。具体的には、ディレクトリD: ¥SOUKOが一杯になったとすると、端末23が”D: ¥SOUKOが一杯です。倉庫をF: ¥SOUKOに変更します。よろしいですか。”等のメッセージを表示装置24の画面に表示する。ユーザAがこれを承認すると、F: ¥SOUKOが新たにSOUKOのディレクトリとなる。所定の容量の空き領域がどのハードディスクにもなかったときは、”残念ながらディスク容量が足りません。ディスクを増設してください。”等のメッセージが表示される。

【0015】次に、端末ソフトの起動時（ホスト計算機21へのアクセス時）に、ハードディスクやメモリの状況等のインストール後に変化した可能性のある情報を取得する（ステップS2）。ここでは、ユーザAのハードディスクがドライブDにあり、空き容量が300Mバイトであることが環境ファイル38に書き込まれる。こうして作成された環境ファイル38の内容は、ユーザAがホスト計算機21にアクセス（接続）したときに、コマンドRIS_SENDEVによりホスト計算機21に送信される（ステップS3）。

【0016】ホスト計算機21は受信した情報をユーザA環境ファイル39として保持する。ユーザA環境ファイル39には、機種、ハードディスク情報HD、格納場所SOUKOのほか、使用しているOS（オペレーティングシステム）とその格納場所が記述されている。ここでは、ユーザAの端末23のOSはWINDOWSであり、その格納場所WINDIRはD: ¥WINDOW Sであることがわかる。

【0017】ホスト計算機21からコマンドRIS_SENDEVに対するレスポンスとしてRIS_SENDEV*RESP OKを受け取ると、端末23はコマンドRIS_KEYLISTにより第1キーリストを要求する（ステップS4）。これに応じて、ホスト計算機21は第1キーテーブル32の内容をRIS_KEYLIST*RESPとともに送り返す。ここでは、第1キーテーブル32には、キー番号1、2、3、・・・に対応するキーワードとして、OS/基本ソフト、開発支援、ゲーム、・・・が格納されている。

【0018】これらのキーワードが第1キーリストとして表示装置24の画面に表示されると（ステップS5）、ユーザAはそれらの中から第1キーワードを選び端末23に入力する（ステップS6）。すると、端末23はユーザAの選んだ第1キーワードのキー番号とともに、第2キーリストを要求するコマンドRIS_KEYLISTをホスト計算機21に送る（ステップS7）。ここでは、ユーザAは第1キーワードとしてゲームを選

択し、それに対応するキー番号3がホスト計算機21に送られる。

【0019】第2キーリストを要求されたホスト計算機21は、受け取ったキー番号に対応して第1キーテーブル32内に格納されているポインタを用いて、対応する第2キーテーブル33を求め、その内容をRIS_KEYLIST*RESPとともに送り返す。ここでは、第2キーテーブル33には、キー番号51、52、53、・・・に対応するキーワードとして、RPG、アクション、パズル/クイズ、・・・が格納されている。

【0020】第2キーテーブルは第1キーテーブル32内のキーワードに対応して一般に複数設けられており、その数は第1キーテーブル32内のキーワードの数と同じか、またはそれより少ない。後者の場合には、第1キーテーブル32内の2つ以上のキーワードが同じ1つの第2キーテーブルを指すことになる。

【0021】第2キーテーブル内のキーワードが第2キーリストとして表示装置24の画面に表示されると（ステップS8）、ユーザAはそれらの中から第2キーワードを選び端末23に入力する（図30、ステップS9）。すると、端末23はユーザAの選んだ第1および第2キーワードのキー番号とともに、第1および第2キーワードの両方に該当するソフトウェアのリストを要求するコマンドRIS_LISTをホスト計算機21に送る（ステップS10）。ここでは、ユーザAは第2キーワードとしてアクションを選択し、それに対応するキー番号52がホスト計算機21に送られる。

【0022】ソフトウェアのリストを要求されたホスト計算機21は、第1および第2キーワードの2つのキー番号を持つソフトウェアをソフトウェア群35の中から検索する。このとき、検索条件として第1キーワードと第2キーワードとを区別せずに、フラットに検索を行う。また、機種やOSの種別はデフォルトのキーとして扱い、これらも加味した上で検索する。これにより、例えばTOWNS以外の機種専用のソフトウェアが検索されてくることが防止される。

【0023】そして、該当するソフトウェアの名称と番号のリストをRIS_LIST*RESPとともに端末23に送る。ここでは、キー番号3と52を持つテトリス、パチンコ等のソフトウェアが該当するので、それらの名称がそれぞれのソフトウェア番号5、30等とともに端末23に送られる。

【0024】ソフトウェアのリストが表示装置24の画面に表示されると（ステップS11）、ユーザAはそれらの中から希望するソフトウェアを選び、端末23に入力する（ステップS12）。すると、端末23はユーザAの選んだソフトウェアの番号とともに、ユーザAの環境がそのソフトウェアの動作に適するかどうかのチェックを要求するコマンドRIS_CHKENVをホスト計算機21に送る（ステップS13）。ここでは、ユーザ

10

20

30

40

50

Aはテトリスを選択し、それに対応するソフトウェア番号5がホスト計算機21に送られる。

【0025】ユーザAが選択したソフトウェアの番号を受け取ったホスト計算機21は、その番号に対応するソフトウェアの動作環境とユーザAの端末23の環境との整合性を調べるためのチェックスクリプト40を用意し、環境チェックを行う。このチェックはチェックスクリプト40の実行プログラムと端末23の端末ソフトとの間のやりとりにより自動的に行われるので、ユーザAは環境チェックが行われていることを必ずしも意識する必要はない(ステップS14)。ユーザAに何らかの問い合わせを行う必要が生じたときにのみ、ホスト計算機21がその問い合わせを行う。

【0026】ここでは、ユーザAが選択したテトリスの動作環境として、OSがWINDOWS、機種がTOWNS、PC98等、推奨ディレクトリ(DIR)名がTETであることが記述されている。これに対して、ユーザA環境ファイル39には、機種がTOWNS、OSがWINDOWSと記述されており、両者を比較することによって機種とOSが適合していることがわかる。

【0027】次に、テトリスのチェックスクリプト40を見るとユーザA側の格納場所WINDIRにVBRJP200.DLLというファイルがあるかどうか調査するためのコマンド"ST4 @WINDIR@VBRJP200.DLL"があるので(MQ1)、ホスト計算機21はこれをRIS_CHKENV*RESPとともに端末23に送る。このとき、ホスト計算機21はユーザA環境ファイル39を参照して、@WINDIR@をD:\WINDOWSに置き換えて送る。ここで、@はワイルドカードを表す。また、ファイルVBRJP200.DLLはテトリスの動作に必要なファイルの1つである。

【0028】このコマンドを受け取った端末23は、ドライブDのディレクトリWINDOWSにファイルVBRJP200.DLLがあるかどうか調べ、その結果をANSとしてホスト計算機21に送り返す。ここでは、該当するファイルがなかったためANS=OFFが送り返される。

【0029】端末23にファイルVBRJP200.DLLがないことを知ったホスト計算機21は、チェックスクリプト40に従って(MQ2)、"VBRJP200をコピーしてよいか?"という問い合わせを端末23に送り、この問い合わせが表示装置24の画面に表示される。ユーザAは表示された問い合わせに対する回答を入力し、端末23がその回答をホスト計算機21に送り返す。ここでは、ANS=はい が送り返され、ホスト計算機21はチェックスクリプト40に従って、リモートインストールを承諾し(RIS=OK)、VBRJP200.DLLのコピーを指示するフラグF2をONにする(MA2)。

【0030】もし、ファイルVBRJP200.DLLが端末23の指定されたディレクトリにあった場合はANS=ONが送り返されるので、その時点でRIS=OKとなる(MA1)。

【0031】このように環境チェックを自動的に行うことにより、ユーザAの環境に適合しないソフトウェアが配送されるのを防ぐことができる。例えば、あるパッケージソフトウェアを通信回線22を介して購入した後、特定のドライバがないとそれが動作しないことを知るといったような事故が未然に防止される。

【0032】RIS=OKとなるとホスト計算機21は環境チェックを終了し、判定結果(JUDGE=OK)とともに、配送先のディレクトリSOUKODIRを端末23に送る。このSOUKODIRは、ユーザA環境ファイルに格納されているSOUKOのディレクトリであるD:\SOUKOの下にテトリスの推奨ディレクトリであるTETをサブディレクトリとして付加した形式で指定される。

【0033】このとき同時に、インストールの可否(RIS)、インストールプログラム(インストーラ)のアイコン登録の有無(ICON)、およびダウンロードの可否(DLOAD)が端末23に送られる。これらのフラグRIS、ICON、DLOADにより、ホスト計算機21はインストール、インストーラのアイコン登録、ダウンロードのうちのどれが可能かを端末23に通知する。

【0034】インストールとはユーザAの選んだソフトウェアを端末23のシステム、例えばWINDOWSに登録して、端末23上で使用可能にすることを意味する。したがって、この場合はそのソフトウェアの実行ファイルをWINDOWS上でアイコン登録する作業までを含む。これに対して、インストーラのアイコン登録とはインストールを実行するプログラムを端末23上でアイコン登録することを意味する。

【0035】ここでは、インストールとダウンロードが許諾され(RIS=OK、DLOAD=OK)、インストーラのアイコン登録は行わない(ICON=NG)という条件が提示される。複雑なインストールプログラムを持つソフトウェアの場合には、インストールが許諾される代わりにインストーラのアイコン登録が必要である旨が提示される。また、WINDOWSを搭載している端末からTOS(TOWNSのOS)用のアプリケーションを要求されたような場合には、ダウンロードのみが許諾される。

【0036】次に、端末23の端末ソフトはインストール、インストーラのアイコン登録、ダウンロードの順に優先順位をつけて、より優先順位の高いものをデフォルトとして設定し、表示装置24の画面に表示する。ここでは、ホスト計算機21により許諾されたインストールとダウンロードのうち優先順位のより高いインストール

がデフォルトとして設定され、インストール方法選択ウィンドウに表示される。

【0037】図32は、インストール方法選択ウィンドウの表示例を示している。図32において、「システム登録」がインストールに相当し、これがデフォルトで選択されている。

【0038】ユーザAは表示されたインストール方法を確認して、確認した旨を入力する(ステップS15)。また、ユーザAはここで表示された設定を変更することもできる。例えば、インストーラのアイコン登録を行いたいときは、図32のインストール方法選択ウィンドウ内の「インストーラのアイコン登録」を選択して入力する。

【0039】基本的には、ユーザAは手間をかけずにできあいのインストールを行いたい場合は「システム登録」を選択し、細かいインストール設定を自分で行いたい場合は「インストーラのアイコン登録」を選択し、格納場所を後で変更したい(別の機種種の端末にインストールしたい)場合は「ダウンロード」を選択する。「ダウンロード」を選択すれば、端末23の機種とは異なる機種用のソフトウェアを入手して動作するかどうか試してみることも可能になる。

【0040】次に、端末23はホスト計算機21から指示された宅配用のサブディレクトリD:¥SOUKO¥TETを、ハードディスク25内に自動的に生成する(ステップS16)。ここでもし、端末23にサブディレクトリD:¥SOUKO¥TETが既に存在している場合は、例えばD:¥SOUKO¥TET 001というサブディレクトリをつくり、これも既に存在している場合はD:¥SOUKO¥TET 002というサブディレクトリをつくる。

【0041】テトリスのファイル本体41はファイルTET1.LZH(F1)とVBRJP200.DLL(F2)とから成り、TET1.LZHは4つのファイルTETRIS.EXE、TOWNS.DRV、PC98.DRV、およびMAC.DRCを圧縮してできている。TET1.LZHを圧縮前の状態に伸長(解凍)するとこれらの4つのファイルに分かれるが、TET1.LZHの解凍はホスト計算機21から端末23に配送された後に行われる。

【0042】宅配用のサブディレクトリを生成した端末23は、リモートインストールの開始を依頼するコマンドRIS_INSTALLを選択したソフトウェアの番号とともにホスト計算機21に送る(図31、ステップS17)。これを受けて、ホスト計算機21は送られた番号に対応するソフトウェアのリモートインストールを開始する。リモートインストールは、ホスト計算機21が作成したテトリスのインストールスクリプト42に従って、ホスト計算機21と端末23の間のやりとりにより自動的に行われる(ステップS18)。

【0043】インストールスクリプト42には、まずファイルTET1.LZHをユーザA側の格納場所@SOUKO@にダウンロードすることを指示する記述がある。そこで、ホスト計算機21は@SOUKO@をSOUKODIR=D:¥SOUKO¥TETに置き換えて、ハードディスク25のサブディレクトリD:¥SOUKO¥TETにTET1.LZHをダウンロードする。

【0044】端末23からダウンロードの完了(OK)を通知されると、次にホスト計算機21は、@WINDIR@をD:¥WINDOWSに置き換えて、ハードディスク25のディレクトリD:¥WINDOWSにVBRJP200.DLLをダウンロードする。

【0045】端末23からダウンロードの完了(OK)を通知されると、次にホスト計算機21は、格納場所@SOUKO@(D:¥SOUKO¥TET)にダウンロードしたTET1.LZHを解凍する指示、LHA X D:¥SOUKO¥TET¥TET1.LZHを送る。これを受けて、端末23はTET1.LZHを前述した4つのファイルTETRIS.EXE、TOWNS.DRV、PC98.DRV、およびMAC.DRCに解凍する。これらの4つのファイルはTET1.LZHと同じサブディレクトリD:¥SOUKO¥TETに保持される。

【0046】端末23から解凍の完了(OK)を通知されると、次にホスト計算機21は、格納場所@SOUKO@(D:¥SOUKO¥TET)の機種@.DRVというファイルを格納場所@WINDIR@(D:¥WINDOWS)に移動させてファイル名をFONT.DRVに変更する指示、MOVE D:¥SOUKO¥TET¥TOWNS.DRV D:¥WINDOWS¥FONT.DRVを送る。このとき、ホスト計算機21はユーザA環境ファイル39を参照して、機種@をTOWNSに置き換えて送る。これを受けて、端末23はサブディレクトリD:¥SOUKO¥TETのファイルTOWNS.DRVをディレクトリD:¥WINDOWSに移動し(ファイル移動)、FONT.DRVというファイル名に変更する(リネーム)。

【0047】端末23からファイル移動およびリネームの完了(OK)を通知されると、次にホスト計算機21は、ファイルTETRIS.EXEのアイコン登録を行う指示、ICON TETRIS.EXEを送る。これを受けて、端末23はサブディレクトリD:¥SOUKO¥TETのファイルTETRIS.EXEをアイコン化して端末23内に登録する。これにより、表示装置24の画面に表示された倉庫ウィンドウ36内に、例えばTETRIS.EXEを起動するアイコン37が表示され、アイコン37をクリックすればテトリスが動作を開始する。

【0048】端末23からアイコン登録の完了(OK)

を通知されると、ホスト計算機21はRETURNを送り返してリモートインストールの終了を端末23に通知し、一連のインストール作業を終了する。リモートインストールの終了を通知された端末23は、ユーザAの指示に従って次のソフトウェアの選択とそのリモートインストールを行うか、あるいは処理を終了する(ステップS19)。

【0049】ステップS18のインストール時に、インストールするソフトウェアがそのダウンロード先の格納場所の空き容量に比べて大きければ、格納場所を変更してダウンロードする。

【0050】次に、図33から図38までを参照しながら、リモートインストールのプロトコルの例と第1および第2のキーワードの表示例について説明する。図33は、ステップS3の端末23によるホスト計算機21へのアクセス時に、端末23が環境情報を送るプロトコルの一例を示している。図33のプロトコルにおいて、リクエストID(RID)、端末を特定するマシンID(MID)、日時(TIME)に続いて、端末23に関するマシン情報(MACHINE:)が記述されている。

【0051】MACHINE:の中のドライブ情報(DRV:)には、容量やドライブ名等のハードディスク25に関する情報が記述されている。例えば、PARTINF:の中のCAPACITYはそのパーティションの容量を表し、VACANTはそのうちの空き容量を表し、DRVNAMEはそのパーティションに対応するドライブ名を表す。

【0052】DRV:に続いて、宅配用の格納場所である倉庫ディレクトリSOUKODIRと、OS(WINDOWS)の格納場所であるディレクトリWINDIRが記述されている。ここでは、SOUKODIR=D:≠RIS≠SOUKOであり、WINDIR=D:≠WINDOWSである。その次には、メモリに関する情報(MEM:)がある。このSOUKODIRを指定することにより、端末23はホスト計算機21に宅配先を通知する。

【0053】そして、MACHINE:に続いて、端末23のマシンパスワード(MPSWD)等が記述される。図34は、ステップS14の環境チェックの終了時に、ホスト計算機21がSOUKODIRにサブディレクトリを付加して送り返すプロトコルの一例を示している。図34のプロトコルにおいて、RID、環境チェックの判定結果(JUDGE)に続いて、宅配先に関する情報(STRPLACE:)が記述されている。

【0054】STRPLACE:の中のSOFTはステップS12でユーザAが選択したソフトウェアの番号を表し、WORKDIRは作業領域を表し、SOUKODIRはホスト計算機21が指定した宅配先を表す。ここでは、WORKDIRは図33のSOUKODIRと同じ

であり、SOUKODIRは図33のSOUKODIRにサブディレクトリ名FMを付加した形になっている。

【0055】STRPLACE:に続いて、作業領域のサイズWORKSIZEおよび宅配先のサイズSOUKOSIZEが記述されている。図35は、ステップS4からS9において、ホスト計算機21が第1および第2キーワードを端末23に送り、ユーザAの選択したキーワードのキー番号を端末23が送り返すプロトコルの一例を示している。図35のプロトコルにおいて、端末23がコマンドRIS_KEYLISTにより第1キーワードを要求すると、ホスト計算機21がレスポンスとしてRIS_KEYLIST*RESPとともに第1キーテーブルの内容KEYLIST:を送り返す。このKEYLIST:内には、キー番号KEY=1、2、3、...に対応してキーワードNAME="OS/基本ソフト"、"開発支援"、"ゲーム"、...が記述されている。

【0056】次に、端末23がコマンドRIS_KEYLISTとともに、ユーザAの選択した第1キーワード"ゲーム"のキー番号3を送ると、ホスト計算機21がレスポンスとしてRIS_KEYLIST*RESPとともに、キー番号3に対応する第2キーテーブルの内容KEYLIST:を送り返す。このKEYLIST:内には、キー番号KEY=51、52、53、54、55、...に対応してキーワードNAME="RPG"、"アクション"、"パズル/クイズ"、"シミュレート"、"冗談"、...が記述されている。

【0057】そして、端末23はコマンドRIS_LISTとともに、ユーザAの選択した第2キーワード"アクション"および"冗談"のキー番号52および55をホスト計算機21に送り、3つのキー番号3、52、55を持つソフトウェアのリストを要求する。ここでは、既におくった第1キーワードのキー番号3はホスト計算機21が記憶しているので、第2キーワードのキー番号52および55のみが送られる。

【0058】図36は、第1および第2キーワードの表示例を示している。図36における第1および第2キーワードは図35のそれらとは異なっている。ユーザAが見ている表示装置24の画面には、例えば画像、ゲーム等の第1キーワードが表示され、ユーザAが第1キーワードを選択すると、次にツール、テキスト、DOS、WIN、画像、音声、ゲーム等の第2キーワードが表示される。第2キーワードの中には、画像、ゲームのように第1キーワードと重複するものも含まれる。

【0059】ユーザAは、表示されたいずれのキーワードも選択することができる。例えば、第1キーワードとして画像を選び、第2キーワードとしてツール、DOS、ゲーム等を選んだり、あるいは第1キーワードとしてゲームを選び、第2キーワードとしてDOS、WIN、画像、音声等を選ぶ。また、ユーザAは第2キーワ

ードとして2つ以上のキーワードを同時に選択することもできる。

【0060】一般に、ソフトウェア群35の中から希望するソフトウェアを選択するためのキーワードは多数あるので、これらを一度に全部表示すると見づらく、まったく必要のないものも表示される。そこで、これを単純なツリー構造のメニューにして表示すると、最初に選択するキーワードが重要な役割を果たし、そこで選択を誤ると欲しいソフトウェアは得られなくなってしまう。

【0061】そこで、図36のように重複を許して2つの階層でキーワードを表示することにより、ユーザは多数のキーワードの中から必要なものをより柔軟に選ぶことが可能になる。尚、キーワードの階層は2階層に限らず、より多くの階層に分けて表示してもよい。

【0062】ソフトウェア群35の各ソフトウェアのコンテンツは、あらかじめいくつかのキーワードと関係付けられて格納されている。例えば図36において、ソフトAはゲーム、画像、DOSの3つのキーワードを持ち、ソフトBは画像、ツール、WIN、ゲームの4つのキーワードを持つ。したがって、ユーザAが画像とゲームの2つのキーワードを選択すると、これらの2つのソフトウェアを含むリストがホスト計算機21から送られてきて画面に表示される。

【0063】図37は、ステップS3のホストアクセス時において、端末23からホスト計算機21へ送られる環境情報の送信プロトコルの他の例を示している。図37のプロトコルにおいて、マシン情報MACHINE：の中のMODELは、ステップS1において端末ソフトのインストール時に取得した情報で、端末23の機種TOWNSを表す。また、PARTINF：の中のVACANTは、ステップS3のホスト計算機21へのアクセスの前（ステップS2）に取得した情報で、対応するパーティションの空き容量を表す。

【0064】図38は、ステップS14における環境チェックとその後のステップS17におけるインストール開始のプロトコルの一例を示している。図38のプロトコルにおいて、まず端末23はコマンドRIS_CHKENVとともにユーザAの選んだソフトウェアの番号（ソフトコード）SOFT=5をホスト計算機21に送り、環境チェックを要求する。

【0065】ホスト計算機21は、まずVBRJP200.DLLというファイルが端末23のシステムディレクトリにあるかどうか調査するための指示CHKEXE：を、レスポンスRIS_CHKENV*RESPとともに端末23に送る。CHKEXE：には、TAG="VBRJP200.DLL"、コマンドCMD="ST4 D:¥WINDOWS¥SYSTEM¥VBRJP200.DLL"、作業用ディレクトリの指定WORKDIR="D:¥RIS¥KOBUTA"、宅配先のディレクトリの指定SOUKODIR="D:¥RI

S¥KOBUTA"等が記述されている。

【0066】端末23は、ドライブDのディレクトリWINDOWS¥SYSTEMにファイルVBRJP200.DLLがあるかどうか調べ、その結果をRESULT：としてホスト計算機21に送り返す。RESULT：には、対応するTAG="VBRJP200.DLL"とともに調査結果VAL="OFF"が記述されている。これは、ファイルVBRJP200.DLLがシステムディレクトリになかったことを意味する。

【0067】そこで、ホスト計算機21は、選択されたソフトウェアをインストールしてよいかどうかをユーザに問い合わせる指示ASKCHK：を、レスポンスRIS_CHKENV*RESPとともに端末23に送る。ASKCHK：には、TAG="Q1"、表示すべき質問文QUERY="このソフトを実行するためにはインストールしてもよろしいですか？"、および回答のフォーマットANS：が記述されている。

【0068】端末23は、ユーザAが入力した回答をRESULT：として、コマンドRIS_CHKENVとともにホスト計算機21に送り返す。RESULT：には、対応するTAG="Q1"とともに回答結果VAL="OK"が記述されている。これは、ソフトウェアをインストールしてもよいという意味である。

【0069】そこで、ホスト計算機21は、他に動作環境上の障害がなければ、レスポンスRIS_CHKENV*RESPとともに環境チェック結果JUDGE="OK"を端末23に送る。これは、ファイルVBRJP200.DLLがないことを除いて、ソフトコード5番のソフトウェアの動作環境が整っており、インストールが可能であることを意味する。

【0070】環境チェックの結果が"OK"となったので、端末23はコマンドRIS_INSTALLとともに、インストール方法の種別TYPE="RIS"、宅配先の指定STRPLACE：等をホスト計算機21に送り、インストールを要求する。STRPLACE：には、インストールするソフトウェアのソフトコードSOFTとともに、作業用ディレクトリWORKDIR、宅配先のディレクトリSOUKODIRが記述されている。

【0071】以後、ホスト計算機21は指定された方法でソフトウェアのインストールを行う。インストールの際には、例えば図31のステップS18で説明したようにホスト計算機21から送られるコマンドにより、端末23がファイルを解凍して、コピーして、システムに登録するという逐次作業を行う。これらの作業が完了したかどうかは、端末23がレスポンスとして逐次送り返すので、ホスト計算機21はインストール作業の進行状況を最後まで監視することができる。

【0072】しかしこの方法では、端末23が頻繁にホスト計算機21との通信を行わなければならないので、

通信の効率が悪くなる。そこで、ホスト計算機21がインストール作業のコマンドを記述した設定ファイルを用意し、これを端末23に渡して自動インストールを行わせる方法が考えられる。

【0073】図39は、設定ファイルを用いた自動インストールのフローチャートである。図39のインストール作業は、図30のステップS16におけるサブディレクトリの生成の後に行われる。

【0074】端末23は、まずコマンドRIS_INSTALLをソフトウェアの番号とともにホスト計算機21に送る(ステップS21)。これを受けて、ホスト計算機21は送られた番号に対応するソフトウェアのリモートインストールを開始する。リモートインストールは、ホスト計算機21が作成したテトリスのインストールスクリプト43に従って、ホスト計算機21と端末23の間のやりとりにより自動的に行われる(ステップS22)。

【0075】ホスト計算機21は、まず@SOUKO@をSOUKODIR=D:¥SOUKO¥TETに置き換えて、ハードディスク25のサブディレクトリD:¥SOUKO¥TETにファイルTET1.LZHをダウンロードする。

【0076】端末23からダウンロードの完了(OK)を通知されると、次にホスト計算機21は、@WINDIR@をD:¥WINDOWSに置き換えて、ハードディスク25のディレクトリD:¥WINDOWSにVBRJP200.DLLをダウンロードする。

【0077】端末23からダウンロードの完了(OK)を通知されると、次にホスト計算機21は、@SOUKO@をSOUKO=D:¥SOUKOに置き換えて、端末23が行うべき作業のコマンドを記述したユーザ用設定ファイル44(SETUP.INF)を、ハードディスク25のディレクトリD:¥SOUKOにダウンロードする。このとき、ホスト計算機21はステップS18と同様にユーザA環境ファイル39を参照して、設定ファイルSETUP.INFに含まれる機種、SOUKO、WINDIR等の情報をユーザA用の情報に書き換えて送る。設定ファイルSETUP.INFには、ファイルTET1.LZHを解凍し、ファイルTOWNS.DRVを移動してリネームし、ファイルTETRIS.EXEをシステムに登録する一連の作業が記述されている。

【0078】端末23からダウンロードの完了(OK)を通知されると、次にホスト計算機21は、設定ファイルSETUP.INFの記述に従って自動インストールを行う指示INSTALL D:¥SOUKO¥SETUP.INFを端末23に送る。

【0079】これを受けて端末23は、設定ファイルSETUP.INFの記述に従って自動インストールを行う。端末23は、まずコマンドLOG OFFによりホ

スト計算機21との間の回線を遮断する。これ以降は、通信回線22を使用しないので通信料金もかからない。次に、コマンドLHA X D:¥SOUKO¥TET¥TET1.LZHにより、ファイルTET1.LZHを前述した4つのファイルTETRIS.EXE、TOWNS.DRV、PC98.DRV、およびMAC.DRVに解凍する。

【0080】次に、端末23はコマンドMOVE D:¥SOUKO¥TET¥TOWNS.DRV D:¥WINDOWS¥FONT.DRVにより、ファイルTOWNS.DRVをディレクトリD:¥SOUKO¥TETからD:¥WINDOWSに移動させて、そのファイル名をFONT.DRVに変更する。次に、コマンドICON TETRIS.EXEにより、ファイルTETRIS.EXEをアイコン化して端末23内に登録する。そして、コマンドPOFFにより電源をオフにし、インストール作業を終了する。

【0081】設定ファイルを用いた自動インストールでは、ホスト計算機21が作業終了の最終確認をすることはできないが、通信回数を削減することにより通信回線22の使用効率を高めることができる。また、ソフトウェアの購入前に機種等のインストール条件を解決して、後は自動的にインストールが行われる。

【0082】図40は、他の形式の設定ファイルの一例を示している。図40の設定ファイルにおいて、[DstDir]はファイルの格納先のリストであり、[Files]は格納すべきファイルのリストである。ここでは、ユーザが希望するソフトウェアの実行ファイルSoft2.ExeをディレクトリD:¥RIS¥KOBUTAに格納し、TOWNS用のドライバであるTOWNS.DRVの名称をFONT.DRVに変更してディレクトリD:¥WINDOWS¥SYSTEMに格納することが記述されている。

【0083】設定ファイルの生成時には、[DstDir]は1=SOUKODIR、2=WINDIRと記述されるが、ホスト計算機21が実際のSOUKODIR、WINDIRの情報を得た時点で、1=D:¥RIS¥KOBUTA、2=D:¥WINDOWS¥SYSTEMと書き換えられる。このように、設定ファイルの中の設定条件は動的または選択的に変更することができる。

【0084】以上説明したように、先願によれば、配送センターから通信回線を介して、ユーザの希望するソフトウェアをその端末装置に自動的にインストールすることが可能になる。これを利用して、通信回線を介してソフトウェアをユーザに販売することも可能になる。

【0085】このとき、配送されたソフトウェアは専用のディレクトリに格納されるので保守性が良く、またユーザは配送先のディレクトリ等を指定する文字列を入力する必要がない。

【0086】ユーザは端末装置の画面上で希望するソフトウェアを効率よく選びだすことができ、そのインストールの方法も選択することができる。また、ユーザがソフトウェアの動作環境をチェックする必要がなく、自動的に環境チェックが行われる。

【0087】さらに、ソフトウェアのインストール時に通信回線の使用をできるだけ少なくすることができる。

【0088】

【発明が解決しようとする課題】しかしながら、上述のリモートインストールシステムには次のような問題がある。

【0089】現在、パソコン通信で流通しているソフトウェアには、様々な種類のものがある。例えば、フリーウェアと呼ばれるものは基本的に無料で配布され、シェアウェアと呼ばれるものは機能制限付きで一旦無料で送付された後、所定の代金が送金されれば機能制限が解除されることになっている。また、商品として販売されているものは、基本的に代金と引き換えに送付される。

【0090】配送センターがリモートインストールの機能を利用して、ソフトウェアをユーザに販売するサービスを行う場合、このような多様な販売形態に対応して、代金を確実に受け取れることを保証する機構が必要になる。特にシェアウェアの場合に、代金の受け取りと機能制限の解除をどのようにして処理するかが問題となる。

【0091】本発明は、通信回線を介してソフトウェアを販売するサービスにおいて、配送したソフトウェアの代金を自動的に決裁するソフトウェア代金決裁システムとその方法を提供することを目的とする。

【0092】

【課題を解決するための手段】図1は、本発明のソフトウェア代金決裁システムの原理図である。図1のソフトウェア代金決裁システムは、ユーザの端末装置に通信ネットワークを介してソフトウェアを配送する情報処理システムに設けられ、ソフトウェア情報記憶手段51、価格情報記憶手段52、処理手段53、および課金手段54を備える。

【0093】ソフトウェア情報記憶手段51は、登録ソフトウェアの本体ファイルと管理情報とを含むソフトウェア情報を記憶する。価格情報記憶手段52は、上記登録ソフトウェアの販売形態および価格に関する価格情報を記憶する。

【0094】処理手段53は、第1のユーザから上記登録ソフトウェアの配送要求を受け取った時、価格情報記憶手段52に記憶された上記価格情報に基づいてその登録ソフトウェアの価格を決定し、上記販売形態に応じた配送処理を行う。

【0095】上記価格情報には、フリーウェア、シェアウェア等の登録ソフトウェアの販売形態とその価格とが記述され、処理手段53はこの価格情報を参照して、配送要求を受けたソフトウェアの価格を決定し、適当な配

送処理を行うことができる。例えば、登録ソフトウェアがフリーウェアの場合は第1のユーザの端末に無料でインストールし、商品の場合は指定された代金と引き換えにインストールする。また、シェアウェアの場合は、所定の代金と引き換えに機能制限の解除に必要な処理を行う。こうして、ソフトウェアの代金が自動的に決裁される。

【0096】処理手段53は、上記ソフトウェア情報および価格情報を、例えばソフトウェアの登録を行う第2のユーザの端末からあらかじめネットワークを介して受け取り、それぞれソフトウェア情報記憶手段51および価格情報記憶手段52に登録しておく。

【0097】また、課金手段54は、処理手段53が決定した代金を上記第1のユーザに対して自動的に課金する。これにより、配送センターまたは第2のユーザが有料ソフトウェアの代金を確実に受け取ることが可能になる。

【0098】例えば、図1のソフトウェア情報記憶手段51および価格情報記憶手段52は、実施形態の図2におけるホスト計算機63に付随する記憶装置に対応し、処理手段53および課金手段54は、ホスト計算機63のCPU（中央処理装置）に対応する。

【0099】

【発明の実施の形態】以下、図面を参照しながら本発明の実施形態を詳細に説明する。図2は、実施形態のリモートインストールシステムの構成図である。図2のシステムにおいては、N個の端末61-1、・・・、61-Nが通信ネットワーク62を介してホスト計算機63と結合している。端末61-1、・・・、61-Nは、サービスの対象となる各ユーザの端末装置であり、ホスト計算機63は配送センター側にある。ホスト計算機63は、ネットワーク62から見れば一台に見えるが、実際には、複数の計算機による分散処理も可能である。まず、ネットワーク62を介してホスト計算機63にソフトウェア（コンテンツ）を登録する方法について説明する。ソフトウェアの作者は、作成した本体ファイルとともに登録に必要な情報をアップロードする必要があるが、このとき、アップロードする情報を標準化しておく。ソフトウェア情報をアップロードする具体的な方法としては、標準化され、いくつかに分類された情報に、その内容を示すTAGを付けて送るようにする。情報を標準化することによって、登録作業やチェック作業の自動化が可能となる。また、情報を分割してアップロードすることにより、バグの発見時に必要な部分だけを再送し、再度のアップロードに要する時間を短縮することができる。アップロード情報は、大きく分類して次の5種類となる。[1]CFGファイル（コンフィギュレーションファイル）：ソフトウェアをシステムに登録するための基礎情報（必須の情報）で、次のものを含む。

○データベースに登録するための情報

コンテンツ名称
 コンテンツの前バージョン、現バージョン（必須ではない）
 販売形態（フリーウェア／シェアウェア／市販品）
 価格
 動作可能なハードウェア
 動作可能なOS
 検索キーワード
 ○インストールの本処理の情報
 コンテンツ本体の圧縮方法
 インストールの方法
 ○インストール時の前処理と後処理の情報
 前処理の有無と方法
 後処理の有無と方法
 [2] 説明ファイル：ソフトウェアの概要を書いた説明（必須の情報）
 [3] 本体ファイル：実行に必要なファイルを圧縮して1つにまとめたもの（必須の情報）
 [4] CHKファイル（チェックファイル）：ソフトウェアが動作する環境条件を調査してリモートインストールを修飾するために、特別な言語で書かれたファイルで、環境依存性のあるソフトウェアのみ必要となる。よく使われるものはシステムで提供するが、特殊なものもオプションとしてユーザが作成する。CHKファイルには、例えば次のような情報が含まれる。
 [0100] CHKEXE：オプションで行われる環境チェックのための実行プログラム
 VERSION：DLL（dynamic link library）のバージョン情報のチェックをオプションで行う必要がある場合に作成される。
 [0101] 図30のチェックスクリプト40のような情報をCHKファイルに記述しておくこともできる。
 [5] インストール関連ファイル：リモートインストールの実際の手順を書いたもので、次のような情報が含まれる。
 [0102] INSTALL：実際のインストールの手順書（手順を標準化した場合は不要）
 標準化したインストール処理は以下のようになり、大半のソフトウェアはこの範囲内でインストール可能である。
 [0103] 1. 指定されていれば、前処理を行う。
 2. 圧縮された本体を端末のワーク領域にダウンロードする。
 3. 本体を展開して、適当なディレクトリにファイルをコピーする。
 [0104] 4. 指定があれば、ファイル属性を変更する。
 5. 指定があれば、メニューやアイコン登録を行う。
 6. 指定があれば、後処理を行う。
 [0105] ICON：メニューとかマネージャといわ

れるものに、ソフトウェアを登録するための情報
 FIRST：INSTALLを標準化した場合に、標準インストール処理の前に行う前処理で、例えば、バージョンアップ処理においては旧版のバックアップ処理等を指す。

[0106] LAST：INSTALLを標準化した場合に、標準インストールの後に行う後処理で、例えば、環境変数ファイルCONFIG.SYSの書換え処理等を指す。

10 次に、図3から図10までを参照しながら、ファイルの具体例について説明する。図3から図6まではCFGファイルの例を示しており、図7、8、9、10は、それぞれ説明ファイル、本体ファイル、CHKファイル、およびインストール関連ファイルの例を示している。

[0107] 図3、4、5、6のCFGファイルにおいて、；で始まる行はコメント文であることを表す。図3の[name]はソフト検索用名称を表し、[version]はそのバージョンを表し、[oldversion]はその旧バージョンを表す。また、[type]は販売形態を表し、[price]は価格を表す。ここでは、[type]はSHAREとなっており、シェアウェアであることを示している。[machine]は動作可能なハードウェアを表し、ここではTOWNS、FMR、PC98、PC-ATの4機種が挙げられている。

[0108] 図4において、[os]は動作可能なOSを表し、ここではWIN31Jのみが挙げられている。

[key]は検索用のキーワードを表し、ここでは“中国語”がキーワードとして記されている。[compression]は本体の圧縮形式を表し、ここではLHA方式と記されている。また、[instype]は可能なリモートインストールの方法を表している。「RIS icon.def」は、通常のリモートインストールを行ってアイコンファイルicon.defに登録する方法を表し、「INSTALLER icon2.def」は、インストーラ（INSTALL.EXE）自体をアイコン登録する方法を表し、「DLOAD」はダウンロードする方法を表す。ここでは、これらの方法のいずれもが可能となっている。

40 [0109] 図5において、[souko]はインストール時にソフトウェアを格納する推奨ディレクトリを表し、ここではP1とP2の2つのディレクトリが記されている。

[0110] [source]は圧縮された本体ファイル（アーカイブファイル）の情報を表し、ここではS1とS2の2つのアーカイブファイルの情報が記されている。S1のファイル名はNEWWP.LZHで、展開後（解凍後）の格納ディレクトリは@P1@、展開時のモードはM、ファイル容量は800000バイト、展開後の最低必要容量は2000000バイトである。また、

S2のファイル名はNEWWPLIB.LZHで、展開後の格納ディレクトリは@P2@、展開時のモードはF、ファイル容量は7000バイト、展開後の最低必要容量は13000バイト、重複チェックの対象ファイル名はGSWDDL.DLLである。

【0111】ここで、モードMが指定されていれば、端末側で展開先ディレクトリを変更することができ、モードFが指定されていれば変更不可能である。また、GSWDDL.DLLは、アーカイブファイルNEWWPLIB.LZHの中に圧縮格納されているファイルの1つで、端末上でファイル名の重複がないかどうかをチェックするために記されている。

【0112】[destination]は特殊な展開先ディレクトリを表し、ここではD1として@WINSYSTEM@が記されている。また、[copy]はインストール時に移動させるべきファイル等の情報を表し、ここでは、C1として、S2で指定したアーカイブファイルに含まれるファイルGSWDDL.DLLを、D1で指定したディレクトリに同じファイル名で移動することが記されている。Attrの欄のNは、移動後も属性が不変であることを表す。

【0113】図6において、[first]は前処理の情報を表し、インストールの際にアーカイブファイルのダウンロード前に実行しておきたいコマンドが記される。[last]は後処理の情報を表し、インストールの際にアイコン登録コマンド実行後に実行したいコマンドが記される。

【0114】図7の説明ファイルは、図3から図6までのCFGファイルに記述された新ワープロA2というソフトウェアの概要をユーザに説明するための情報であり、図8の本体ファイルは、そのアーカイブファイルを示している。アーカイブファイルNEWWP.LZHは、3つのファイルWP.EXE、NEWWP.ICO、README.DOCを含んでおり、アーカイブファイルNEWWPLIB.LZHは1つのファイルGSWDDL.DLLのみを含んでいることが分かる。

【0115】図9のCHKファイルは、インストール時の環境チェックのチェックスクリプトに対応する。ここでは、WP.EXEというファイルがユーザの端末にあるかどうかのチェックと、もしあれば、それが最新のバージョンかどうかのチェックと、それぞれのチェック結果に応じた処理とが記述されている。

【0116】図10は、インストール関連ファイルの一種であるICONファイルを示している。図10のICONファイルには、新ワープロA2の実行ファイル、カレント・ワーキング・ディレクトリ、アイコン登録用のファイル等の情報が記述されている。このICONファイルは、リモートインストール時のシステム登録の際に使用される。

【0117】このようなCFGファイル、説明ファイル

ル、本体ファイル、CHKファイル、およびインストール関連ファイルを、端末からホスト計算機63にアップロードすることにより、新ワープロA2が自動的に登録される。登録されたソフトウェアは、ユーザからの要求に応じて有償または無償で端末に配送される。

【0118】ところで、リモートインストールシステムによりソフトウェアを販売する場合、ソフトウェアの種類やユーザの過去の購入履歴等に応じて、課金のための特別な手続きや価格の変更が必要になることがある。そこで、宅配するソフトウェアの種類に応じた課金を行うために、さらに課金に関する手続きを定義したファイルをソフトウェア登録時にアップロードしておく。

【0119】図11は、端末61-i (i=1, 2, ..., N) 上で作成されるファイル群の例を示している。図11においては、メモリ71内で、CFGファイル72、説明ファイル73、本体ファイル74、CHKファイル75、インストール関連ファイル76に加えて、定義ファイル77および書換えファイル78が作成される。そして、これらのファイル群がホスト計算機63にアップロードされる。

【0120】ここで、シェアウェアソフトの作者がそれをホスト計算機63に登録する場合を考える。シェアウェアの場合には、作者は、シェアウェアの送金手続き等を定義ファイル77に記述しておく。定義ファイル77には、ソフトウェア名、ソフトコード、送金先、送金金額、送金のタイプ、ユーザシステムのアップデートの方法が記述される。

【0121】図12は、シェアウェア用の定義ファイル77の例を示している。図12の定義ファイルAAAS.CFGにおいて、[name]はシェアウェア「AAスケジューラ」の名称を表し、[type]には送金手続きが記述されている。ここでは、ソフトコード555に対する送金先のIDとして、FJOKIが記されている。AAAは指定追番のラベルを表し、ここでは「AAスケジューラ」を意味している。[price]は価格を表し、[machine]と[os]は、それぞれ動作可能な機種とOSを表し、[key]は検索用のキーワードを表す。

【0122】[instype]には送金のタイプが記述され、ここでは、送金依頼されればその場で機能制限を解除する処理が記されている。また、[last]にはインストールの後処理として、ユーザファイルのアップデート処理が記述される。ここでは、コマンドCHGINIを用いて、ファイルINI.DEFに記述されている内容に従いユーザのファイルを書き換える処理が記されている。

【0123】尚、通常のソフトウェアとシェアウェアとの送金手続きの違いは、[type]のセクションで識別される。CHKファイル75は、送金手続きに関するオプションファイルの1つとしても使用され、インスト

ール時にユーザとホスト計算機63がインタラクティブにやりとりする内容が記述される。図13は、このようなCHKファイル75の例を示している。図13のCHKファイルAAAS. CHKには、ファイルAAA. INIがユーザシステムにあるかどうかを調べて、その格納場所をユーザに確認する処理が記述されている。AAA. INIはシェアウェアの機能を設定するためのファイルで、その[User Info]のセクションに正しい情報を書込まないと、シェアウェアは完全には動作しない。

【0124】書換えファイル78は、ユーザシステムの書換え方法を定義するオプションファイルである。図14は、書換えファイル78の例を示している。図14の書換えファイルINI. DEFには、ユーザシステムのファイルAAA. INIの中の[User Info]のセクションのLicenceの値をAA123AAAに書換える処理が記述されている。Licence=AA123AAAは、シェアウェアの機能制限を解除する暗号コードの役割を果たす。

【0125】図15は、図11に示したファイル群のアップロード処理の例を示している。図15において、シェアウェアの作者は、まず本体登録ファイルとして、CFGファイル72 (AAA. CFG)、説明ファイル73 (AAA. TXT)、インストール関連ファイル76 (ここではアイコンファイルICON. DEF)、および本体ファイル74 (AAA. LZH)を自分の端末からアップロードする。ホスト計算機63は、これらのアップロード情報をコンテンツデータベース81に登録する。ここでは、アップロードされたソフトウェア「AAAスケジューラ」のソフトコード、名称、タイプ(TYPE)、本体ファイル名、説明ファイル名、アイコンファイル名等が管理情報として登録されている。タイプの欄のSHAREはソフトウェアの種類がシェアウェアであることを表す。コンテンツデータベース81は、例えば、ホスト計算機63内または外部のディスク装置内に設けられる。

【0126】次に、作者は送金手続きファイルとして、定義ファイル77 (AAAS. CFG)、CHKファイル75 (AAAS. CHK)、および書換えファイル78 (INI. DEF)をアップロードする。これにより、送金手続きファイルAAASのソフトコード、名称、タイプ、CHKファイル名、後処理用のファイル名等がコンテンツデータベース81に登録される。後処理用のファイルとして、ここでは、書換えファイル78のファイル名INI. DEFが記されている。また、タイプの欄のSOKIN#RISは、ソフトウェアの種類がシェアウェアの送金手続きファイルであることを表し、AAAS. CFGの[instype]の情報に対応している。

【0127】次に、図16から図20までを参照しながら、

ら、アップロードされたファイル群を用いたシェアウェアの送金手続きについて説明する。シェアウェア手続きにおいては、先願のリモートインストールの手続きに加えて、プロトコル上に送金フラグを設ける。そして、端末からこのフラグを立ててソフトウェアの検索を要求することにより、シェアウェア手続きを選択できるようにする。また、端末の画面に送金フラグをON/OFFするメニューを表示させる。

【0128】図16、17、18、19は、シェアウェア「AAAスケジューラ」の送金手続きを示している。ただし、送金手続きの前に、シェアウェア「AAAスケジューラ」は機能制限付きでユーザシステムにインストールされているものとする。ユーザがこのソフトウェアを購入しようとした時は、ホスト計算機63と端末の間で図示のようなコマンド/レスポンスのやりとりを行う。

【0129】図16において、端末は、まずユーザ環境をホスト計算機63に送信し(ステップS31)、ホスト計算機63はこれを受信すると応答を返す(ステップS32)。次に、ユーザがソフトウェア検索用のキーワードリストを要求すると(ステップS33)、ホスト計算機63はキーワードリストを返送する(ステップS34)。

【0130】このとき、図17に示すように、端末の画面上にはオプション手続きのメニュー82が表示され、ユーザはそこから「送金」を指定し、キーワードリストの中からキーワードを選択する。これにより、送金フラグSOKINが立てられ(オンになり)、シェアウェアの検索を開始する指示がホスト計算機63に送られる(ステップS35)。ホスト計算機63は、指定されたキーワードでコンテンツデータベース81を検索し、タイプがSOKINで始まるソフトウェアの名称とその送金手続きファイル(送金ソフト)のソフトコードとを返す(ステップS36)。ここでは、「AAAスケジューラ」、「BBBスケジューラ」等の名称および送金ソフトのソフトコード4000、4001等が返送されている。

【0131】端末は送金ソフトのみリスティングし、ユーザはリスティングされた中の特定の送金ソフトを指定する(ステップS37)。ここでは、ソフトコード4000が指定されている。ホスト計算機63は、ソフトコード4000の送金ソフトの条件により端末側とネゴシエーションを行う(ステップS38)。ここでは、まず、コマンドST4を用いてユーザシステム内のAAA. INIの位置を調べるように端末に指示する。これを受けて、端末はAAA. INIの場所を調べ、その場所はE: ¥AAAということをもホスト計算機63に通知する(ステップS39)。ST4というコマンドは、あらかじめ端末側に具備されているものとする。

【0132】次に、図18において、ホスト計算機63

はダイアログBOX 83を端末の画面に表示させ、AAA. INIの位置はE: ¥AAAでいかにどうかをユーザに確認する(ステップS40)。ダイアログBOX 83に表示されたディレクトリが正しければ、ユーザはそのディレクトリをそのまま返送し(ステップS41)、ホスト計算機63は、ユーザシステムの書き換えるべきファイルのディレクトリパスはE: ¥AAA ¥AAA. INIと確定する。もし、表示されたディレクトリが正しくなければ、ユーザは正しいディレクトリ名を入力する。例えばG: ¥GGGと入力すると、端末はディレク

トリパスG: ¥GGG ¥AAA. INIをホスト計算機63に返す。AAA. INIの格納場所が確定したので、ホスト計算機63は、シェアウェア送金が可能であることを端末に通知する(ステップS42)。
【0133】次に、図19において、ユーザは機能制限の解除を要求する(ステップS43)。これを受けて、ホスト計算機63は図12の定義ファイルの[instype]を参照し、代金をそのユーザの口座等から引き落とした後に、AAA. INIの書換え手順が記述されている書換えファイルINI. DEFを送付する。さらに、定義ファイルの[last]を参照して後処理のコマンドCHGINIを送り、INI. DEFの手順に従って書換えを行うことを端末に指示する。

【0134】これを受けて、端末はダウンロードされたINI. DEFを参照し、図20に示すようにAAA. INIを書換える。図20においては、書換え後の[User Info]のLicenceにAA123AAAが書込まれていることが分かる。これにより、シェアウェア「AAAスケジューラ」の機能制限が解除され、ユーザシステム上で完全に動作するようになる。

【0135】その後、ホスト計算機63はINI. DEFを端末から削除して、処理を終了する。図12の定義ファイルでは、[instype]のセクションにその場で制限解除を行うという情報が記されていたので、代金の引き落としと同時にシェアウェアの機能制限を解除した。しかし、一般のパソコン通信センターと同様に、ホスト計算機63が代金引き落としとシェアウェア登録者への電子メールの発行だけを行う構成とすることもできる。

【0136】図21は、代金引き落としおよび電子メールの発行のみをサポートする場合の定義ファイル77の例を示している。図21の「BBBスケジューラ」の定義ファイルBBBS. CFGにおいて、[name]から[key]までのセクションについては図12と同様である。[instype]には送金処理のみが記されており、その場での機能制限の解除は行われない。また、この場合は、「BBBスケジューラ」の作者は図13、14のようなCHKファイル、書換えファイルを用意しなくてもよい。定義ファイルBBBS. CFGがホスト計算機63にアップロードされると、コンテンツデ

ータベース81のタイプの欄にはSOKIN#MAILと記録される。

【0137】図22、23、24は、ソフトウェア「BBスケジューラ」の代金引き落とし手続きを示している。ただし、この手続きの前に、シェアウェア「BBスケジューラ」は機能制限付きでユーザシステムにインストールされているものとする。ユーザがこのソフトウェアを購入しようとした時は、ホスト計算機63と端末の間で図示のようなコマンド/レスポンスのやりとりを行う。

【0138】図22において、端末は、まずユーザ環境をホスト計算機63に送信し(ステップS51)、ホスト計算機63はこれを受信すると応答を返す(ステップS52)。次に、ユーザがソフトウェア検索用のキーワードリストを要求すると(ステップS53)、ホスト計算機63はキーワードリストを返送する(ステップS54)。

【0139】このとき、図23に示すように、端末の画面上にはオプション手続きのメニュー82が表示され、ユーザはそこから「送金」を指定し、キーワードリストの中からキーワードを選択する。これにより、送金フラグSOKINが立てられ、シェアウェアの検索を開始する指示がホスト計算機63に送られる(ステップS55)。ホスト計算機63は、指定されたキーワードでコンテンツデータベース81を検索し、タイプがSOKINで始まるソフトウェアの名称とその送金ソフトのソフトコードとを返す(ステップS56)。端末は送金ソフトのみリスティングし、ユーザはリスティングされた中からソフトコード4001の送金ソフトを指定する(ステップS57)。

【0140】次に、図24において、ホスト計算機63はメッセージ84を端末の画面に表示させ、課金していかどうかをユーザに確認する(ステップS58)。このとき、ホスト計算機63は図21の定義ファイルの[instype]を参照し、フラグSOKINの値をメール発行のみを表す「0x08」に変更して、端末に返送する。ユーザは、課金されてもよければOKを、購入しない場合はNGを選択する。ここでは、OKが選択され、登録者に対するメールの発行の依頼が端末からホスト計算機63に送られる(ステップS59)。これを受けて、ホスト計算機63は代金をそのユーザの口座等から引き落とし、「BBスケジューラ」の登録者に代金引き落としを知らせる電子メールを送る。メールの送付先としては、定義ファイルの[type]のセクションの送金先FJOKIを用いる。

【0141】ホスト計算機63から電子メールを受け取った登録者は、電子メール等の手段により、ソフトウェアの購入者に機能制限解除の方法を連絡する。これにより、購入者は「BBスケジューラ」のすべての機能を使用することができるようになる。

【0142】ところで、ユーザがあるソフトウェアの旧バージョンまたは関連するソフトウェアを既に購入しているかどうかによって、そのユーザに対する販売価格の設定を変えたい場合もあり得る。この場合、ソフトウェアの登録者は、定義ファイル77の[price]のセクションに、複数の価格とその販売条件とを記述しておく。例えば、既購入のソフトウェアのソフトコード、ユーザID (UID)、およびマシンID (MID) とを用いて、価格の変更条件を記述することができる。ここで、UIDはユーザを一意に特定する識別子であり、MIDはインストール先の端末を一意に特定する識別子である。このような、定義ファイル77を用いた複数の価格設定の方法は、シェアウェアの送金手続きに限らず、通常の商品として販売されるISV (independant software vender) ソフトの価格設定にも同様に適用できる。

【0143】図25は、複数の価格を設定した定義ファイル77の例を示している。図25の「CCCスケジューラ」の[type]のセクションは、ISVソフトであることを表す。また、[price]のセクションには、0円、2000円、5000円の3つの価格と、それらのうちから販売価格を選択するための条件が記されている。販売価格の選択条件は次の通りである。

(a) アクセスしてきたユーザおよび端末がソフトコード111のソフトウェアXSOF Tを既に買っている場合は無料。XSOF Tは、例えば、CCCスケジューラの前身となるソフトウェアである場合もあり、CCCスケジューラの旧バージョンである場合もある。

(b) アクセスしてきたユーザまたは端末がソフトコード123または124のソフトウェアYSOF Tを買っている場合は2000円。YSOF Tは、CCCスケジューラの関連ソフトウェアである。

(c) (a) または (b) に該当しない場合は5000円。また、ホスト計算機63のデータベースは図26に示すように構成される。図26において、カスタマデータベース85は、リモートインストールシステムのユーザとその端末に関する管理情報を記憶し、購入データベース86は、過去にソフトウェアを購入したユーザ等に関する購入履歴を記憶する。また、コンテンツデータベース81のCCCスケジューラのPRICEの欄には、価格の選択条件を記した価格ファイル87の格納先が記録される。これらのデータベース85、86、81は、ホスト計算機63内または外部のディスク装置内に設けられ、価格ファイル87もまたそのディスク装置に格納される。ホスト計算機63は、コンテンツデータベース81を参照してCCCスケジューラの価格ファイル87を読み出し、その記述に従って価格を決定する。

【0144】図27は、価格ファイル87に基づく価格決定処理のフローチャートである。図27において処理が開始されると、ホスト計算機63は、まずカスタマデ

ータベース85と購入データベース86を参照し、アクセスしてきたユーザのUIDおよびその端末のMIDの組みあわせで、ソフトコード111のソフトウェアが購入されたことがあるかどうかを調べる(ステップS61)。もし、そのような履歴データがあれば価格を0円として(ステップS62)、処理を終了する。そのような履歴データがなければ、次に、アクセスしてきたユーザのUIDまたはその端末のMIDを含む購入履歴の中に、ソフトコード123または124のソフトウェアの購入記録があるかどうかを調べる(ステップS63)。もし、そのような履歴データがあれば価格を2000円として(ステップS64)、処理を終了する。そして、そのような履歴データがなければ、価格を5000円として(ステップS65)、処理を終了する。

【0145】今、CCCスケジューラの購入を希望するユーザのUIDをFJOTAとし、そのMIDを0005とする。図26の購入データベース86には、そのユーザがその端末を用いてソフトコード111のソフトウェアを購入したことが記録されているので、ステップS61の判定結果はYESとなり、CCCスケジューラは無料で宅配される。

【0146】もし、ソフトコード111、123、124のソフトウェアのいずれも購入していないユーザが、それらのいずれも購入していない端末からアクセスした場合には、価格は最高額の5000円となる。また、ソフトコード111の購入履歴があったとしても、そのUIDとMIDのうち片方がアクセスに用いられた識別子と一致しなければ、ソフトウェアは有料となる。

【0147】図25の定義ファイルは価格設定の一例に過ぎず、登録者はソフトコード、UID、MIDを組み合わせて、他の任意の選択条件を設定してもかまわない。例えば、特定のUIDまたはMIDに対しては、特別価格で販売するように設定することもできる。このように、定義ファイル77に価格の設定条件を記述してアップロードすることにより、ソフトウェアの多様な価格設定が可能になる。

【0148】また、販売対象のソフトウェアがISVソフトではなくシェアウェアの場合は、図19または図24における代金引き落としの際に、その価格ファイルに基づいて価格が決定される。

【0149】

【発明の効果】本発明によれば、通信回線を介してソフトウェアを販売するサービスにおいて、多様なソフトウェアの代金を自動的に決定し、課金処理を行うことができる。特に、シェアウェアの場合に、代金を引き落としから機能制限の解除を行うことが可能になる。

【0150】また、ソフトウェアの登録者は代金の決裁方法を指定して、その情報をあらかじめ登録しておくことができる。例えば、フリーウェア、シェアウェア、有料ソフトウェアの区別や、価格の設定条件等が指定され

る。

【図面の簡単な説明】

【図1】本発明の原理図である。

【図2】実施形態のシステム構成図である。

【図3】CFGファイルを示す図（その1）である。

【図4】CFGファイルを示す図（その2）である。

【図5】CFGファイルを示す図（その3）である。

【図6】CFGファイルを示す図（その4）である。

【図7】説明ファイルを示す図である。

【図8】本体ファイルを示す図である。

【図9】第1のCHKファイルを示す図である。

【図10】ICONファイルを示す図である。

【図11】端末上で作成されるファイル群を示す図である。

【図12】第1の定義ファイルを示す図である。

【図13】第2のCHKファイルを示す図である。

【図14】書換えファイルを示す図である。

【図15】アップロードを示す図である。

【図16】シェアウェア手続きを示す図（その1）である。

【図17】シェアウェア手続きを示す図（その2）である。

【図18】シェアウェア手続きを示す図（その3）である。

【図19】シェアウェア手続きを示す図（その4）である。

【図20】ファイルの書換えを示す図である。

【図21】第2の定義ファイルを示す図である。

【図22】代金引き落とし手続きを示す図（その1）である。

【図23】代金引き落とし手続きを示す図（その2）である。

【図24】代金引き落とし手続きを示す図（その3）である。

【図25】第3の定義ファイルを示す図である。

【図26】ホスト計算機のデータベースを示す図である。

【図27】価格決定処理のフローチャートである。

【図28】先願のリモートインストールシステムの構成図である。

【図29】リモートインストールのフローチャート（その1）である。

【図30】リモートインストールのフローチャート（その2）である。

【図31】リモートインストールのフローチャート（その3）である。

【図32】インストール方法選択ウィンドウの例を示す図である。

【図33】リモートインストールプロトコルを示す図（その1）である。

【図34】リモートインストールプロトコルを示す図（その2）である。

【図35】キーワード選択のプロトコルを示す図である。

【図36】キーワードの表示例を示す図である。

【図37】環境情報の送信プロトコルを示す図である。

【図38】環境チェックのプロトコルを示す図である。

10 【図39】自動インストールのフローチャートである。

【図40】設定ファイルの例を示す図である。

【符号の説明】

21、63 ホスト計算機

22 通信回線

23、61-1、61-N、61-i 端末

24 表示装置

25 ハードディスク

31、38、39 環境ファイル

32 第1キーテーブル

20 33 第2キーテーブル

34、44 設定ファイル

35 ソフトウェア群

36 倉庫ウィンドウ

37 アイコン

40 チェックスクリプト

41 ファイル本体

42、43 インストールスクリプト

51 ソフトウェア情報記憶手段

52 価格情報記憶手段

30 53 処理手段

54 課金手段

62 ネットワーク

71 メモリ

72 CFGファイル

73 説明ファイル

74 本体ファイル

75 CHKファイル

76 インストール関連ファイル

77 定義ファイル

40 78 書換えファイル

81 コンテンツデータベース

82 メニュー

83 ダイアログBOX

84 メッセージ

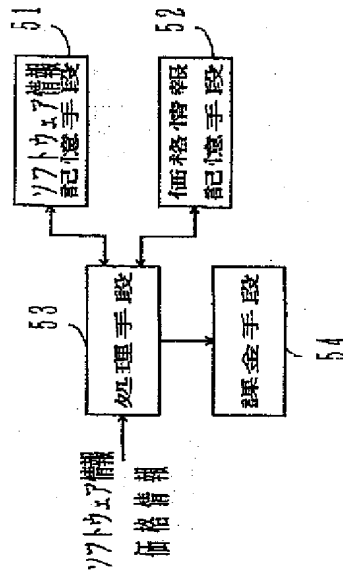
85 カスタマデータベース

86 購入データベース

87 価格ファイル

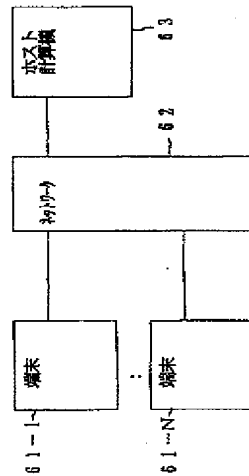
【図1】

本発明の原理図



【図2】

実施形態のシステム構成図



【図8】

NEWWP.LZH=WP.EXE+NEWWP.ICO+README.DOC
 NEWWPLIB.LZH=GSWDDL.DLL

【図13】

第2のCHKファイルを示す図

CHKファイル—AAS.CHK
 MQ1 FILE "AAS.INI"; AAS.INIは必ずあるか
 MA1 "NONE" then DLOAD
 else IC#SOUKO="ST4DIR0";
 MQ2 FILE "WP.EXE";
 MA2 ST4DATE<"1995/06/30 00:00:00" then SQ2;
 else SQ1;
 SQ1 QUERY "貴方のマシンに、既に、最新版が入ってます";
 SA1 "実行" then RIS;
 "不要" then NG;
 SQ2 FILE "GSWDDL.DLL";
 SA2 "NONE" then active=S2, active=C1, RIS;
 else RIS;
 SQ2 QUERY "あなたのシステムにはAAS.INIがありません。";
 SA2 "実行" then NG;

【図7】

説明ファイルを示す図

新ワープロA2は、当社が最新の技術をもって開発した、中国語まで対応出来るワープロです....

【図9】

第1のCHKファイルを示す図

MQ1 FILE "WP.EXE";
 MA1 "NONE" then DLOAD
 else IC#SOUKO="ST4DIR0";
 MQ2 FILE "WP.EXE";
 MA2 ST4DATE<"1995/06/30 00:00:00" then SQ2;
 else SQ1;
 SQ1 QUERY "貴方のマシンに、既に、最新版が入ってます";
 SA1 "実行" then RIS;
 "不要" then NG;
 SQ2 FILE "GSWDDL.DLL";
 SA2 "NONE" then active=S2, active=C1, RIS;
 else RIS;

【図3】

CFGファイルを示す図 (その1)

[name]
:ソフト検索用名称 — 検索DBに登録
:フォーマットは,"名称","読み"とする。名称、読みとも,"'"で括る。
"新ワープロ-A2","シンワープロエーニ"

[version]
:ASCII 8文字以内 — ソフトDBに登録
V1.2

[oldversion]
:旧バージョンソフトの登録
:フォーマットは"ソフトコード バージョン文字列"
V1.1

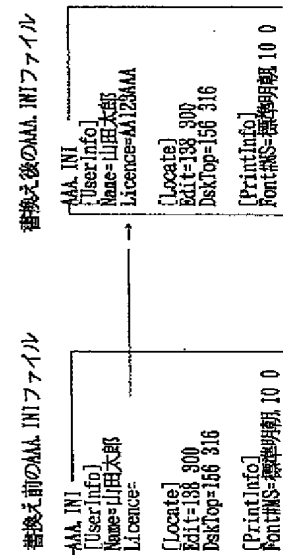
[type]
:製品は、ダウンロード開始時に料金徴収される
:FREE
:フリーウェア
SHARE
:シェアウェア
:PRODUCT
:製品

[price]
:価格を数字のみで記す。頭に'¥'をつけたり、数字を','で区切ったりしてはいけない
3000

[machine]
:フォーマットはマシン識別子 マシン名 — インストールDBに登録
:ホストが必要なのはマシン識別子。マシン名は登録者のために添付
:動作可能マシンのみコメントをはずす
TOWNS
: Townsシリーズ
FMR
: FMRシリーズ
PC98
: PC98シリーズ
PC-AT
: AT互換機
:MAC1
: Quadra 800
:MAC2
: LC50

【図20】

ファイルの書換えを示す図



【図10】

ICONファイルを示す図

[ICON]
GROUP=RIS倉庫
TITLE=新ワープロ-A2
EXE=@IC#SOUK00#WP.EXE
CWD=@IC#SOUK00
WICON=@IC#SUK00#NEWWP.ICO
TEXTTYPE=WIN

【図4】

【図14】

書換えファイルを示す図

CFGファイルを示す図 (その2)

```

[os]
:[machine]と同様      —インストールDBに登録
WIN31J
: 日本語 Windows 3.1
:NT31J
: 日本語 Windows NT 3.1
:DOS5V
: DOS/V Ver. 5.x
:DOS5J
: 日本語MS-DOS Ver. 5.0
:DOS3J
: 日本語MS-DOS Ver. 3.x
:TOS
:TOWNS OS
:MACOS7
: MacOS Ver. 7.0
[key]

:フォーマットは、「キー1」、「キー2」、...、「キーn」を
:一行に書く。複数指定可能      —検索DBに登録
: 中国語
[compression]
: 圧縮の形式を示す。コメントの付け換えのみ可
:NONE
: 圧縮なし
LHA
: LHA による圧縮
: TGZ
: gun tar + gzipによる圧縮
: ZIP
: PKZIP による圧縮
[instype]
: インストールタイプ, 許さないTYPEをコメントアウト
: フォーマットは, "ID [iconfile1 [iconfile2 [...]]]"とする。
RIS      icon.def
: install してシステムに登録
INSTALLER icon2.def
: INSTALL. EXE自体をICON登録
DLOAD
: ダウンロード

```

書換えファイルINI. DEF

<@IC#TARGET@#AAA. INI> : ユーザシステムのAAA. INIファイルを書き換える。
 [User Info] : [User Info] セクションのLicenceの値をAA123AAAにする
 Licence=AA123AAA

【図34】

リモートインストールプロトコルを示す図 (その2)

```

RIS_CLIENT+RESP(ホスト→端末)
RID=6,
JUDGE=1,
STRPLACE:
SOFT=5,
WORKDIR="D:\RISYS\SOUKO",
SOURDIR="D:\RISYS\SOUKO\FPM"  ←サブディレクトリ付
:
WORKSIZ=309000,
SOUKOSIZ=549

```

【図5】

CFGファイルを示す図 (その3)

[souko]

ソフトを格納する推奨ディレクトリを記述する。@SOUKO@で始まることが望ましい。
 フォーマットは、"ID ディレクトリ名"で、"ID"は、"P"で始まり数字が続く("Pn")。

P1 @IC#SOUKO#ABC

P2 @WORK@

[source]

フォーマットは、"ID FILE ExtPath mode arsize extsize extfiles"とする。

ID: S1... Sn の nは連番を記述

FILE: ダウンロードする本体ファイル(LZH形式のはず)名を記述

ExtPath: 展開すべきディレクトリ名を記述。特殊ディレクトリ記述あり

ファイル名はASCIIのみ許す

mode: 端末側で、展開先ディレクトリを変更していいかどうかを示す。

M: 変更可能(Movable) F: 変更不可能(Fixed)

arsize: アーカイブの容量(bytes)

extsize: 展開後の最低必要量(bytes)

インストール時に大量にWork領域を必要とする場合は、arcのサブを読む

実行時に大量にWork領域を必要とする場合は、minのサブを読む

ただし、他のソフトのインストールにより、領域不足となる可能性あり

extfiles: FILEで示されるアーカイブファイルの中に圧縮格納されているファイル名

を、で区切って記述する。全てのファイルを書く必要はなく、端末で重

複チェックを行ないたいものだけ記述すればよい。従って、省略は可能と

する。なお、途中で改行を入れてはいけな。必ず全てのファイルを一行

に記述すること。

ID	FILE	ExtPath	mode	arsize	extsize	extfiles
S1	NEWWP.LZH	@P1@	M	800000	2000000	
*S2	NEWWPLB.LZH	@P2@	F	7000	13000	GSWDL.DLL

[destination]

フォーマットは、"ID ディレクトリ名"で、"ID"は、"D"で始まり数字が続く("Dn")。

ファイルを展開する時の展開先ディレクトリを列挙。基本的に特殊ディレクトリのみ

記述可
 —インストールDBに登録

D1 @WINSYSTEM@

[copy]

移動させるべきファイルや、Mode変更(場所移動なしでも)がある場合のみ記述

[destination]セクションと同時に使用してもよいが、直接移動先ファイル名(特殊

ディレクトリを含むフルパス)を書いてもよい。

フォーマットは、"ID @SID@SourcePath @DID@DestPath Attr"

モードを変更するだけで、移動がない場合のDestinationは、*とする。

ID: 順に C1, C2, ..., Cn とする。チェックスクリプトで参照可能

SID: [source]で記述した ID

SourcePath: 展開後現れるファイル名(サブディレクトリを含んでよい)

DID: [destination]で記述した ID

DestPath: 展開後現れたファイルの移動先ファイル名(サブディレクトリを含んでよい)

Attr: 移動後隠す/書き込み禁止などを実行("attrib +n"のみ実行。"-n"はなし)

R: RO H: Hidden A: Archive S: System N: Noneを続けて記述

何も設定しない時は、省略可

ID	Source	Destination	Attr
*C1	@S2@GSWDL.DLL	@D1@GSWDL.DLL	N

【図6】

CFGファイルを示す図 (その4)

[first]
 インストールの際に、アーカイブファイルのダウンロード前に実行しておきたいコマンドを記述する。例えば、ファイルのリネームなどである。
 形式は、ID TYPE COMMAND ARGS FILES とする。
 ID: "Fn" (n = 1, 2, 3, ...) と記述する。
 頭に '*' をつけた場合は、"Sn", "Cn" と同じでデフォルトでは実行しません。
 チェックスクリプトで ACTIVATE=Fn とした場合のみ実行します。
 TYPE: S, C, D, L のいずれかを記述。それぞれ、S[cript], C[ommand], D[ownload], L[ha] の略である。
 Script: インストールスクリプトに、サブルーチン的に機能を付け加えておき、それを呼び出す。ホスト側で機能を想定して作成しておく。現在は未対応。
 Command: 端末プログラム配布時にある程度想定したコマンドを送りつけておき (SI4, Intresio などと同じ)、それを実行する。もしくは、他のデータなどと一緒に送った LZH ファイルの中に予め仕込んであったものを実行する。
 Download: 個別にプログラムをダウンロードし、実行
 Lha: Download の一種だが、ホスト上では lha で固めてある。FILES 欄には、*.LZH を一つだけ書ける (この固まりにプログラム、データをまとめて入れておく)。
 つまり、この TYPE は、コマンドを実行する際に、そのコマンドが「どこ」にあるかを指定するものである。当面、"C", "D" の利用を想定している。
 COMMAND: 実行すべきコマンドを記述する。注意点として、
 1) TYPE が S, D, L の場合は、コマンド名だけで良い。
 この場合の LZH は、ワークディレクトリに展開されるものとする。LZH で端末にファイルが渡された場合、後始末で *.lzh 以外のファイルを消せない (ホストは lha で何が展開されるかわからない) ので、Fn を使って自分で削除定義をする必要がある。
 2) TYPE が C の場合には、DLOAD による *.LZH に仕込んであった場合は、その展開場所を変数で指定。端末プログラムとして配布済みの場合は、コマンド名だけで良い (パスに記述されてるから)。
 ARGS: COMMAND に与える引数を、" " で括って記述する。変数 (@...@) を使用してよい。引数がない場合は、空文字を " " で括っておく。
 FILES: 必要なファイルが1つとは限らないので、" " で区切って複数記述して良いこととする。全てのファイルのダウンロードが終了した後に、COMMAND を実行する。

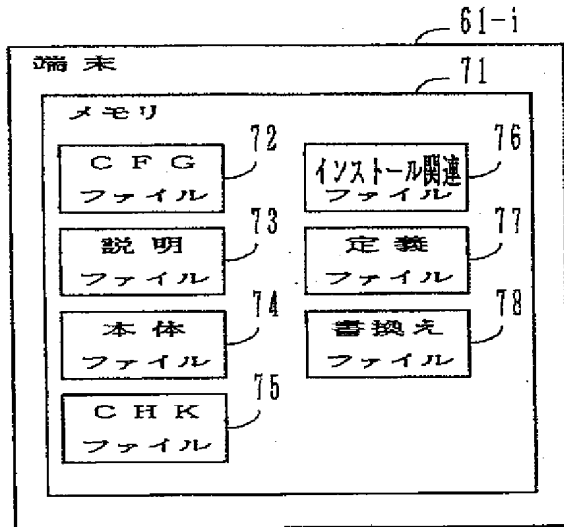
ID	TYPE	COMMAND	ARGS	FILES
FI	C	BACKRIS	"WP.EXE"	BACKRIS.EXE

 [last]
 インストールの際に、アイコン登録コマンド実行後に実行したいコマンドを記述する。
 例えば、AUTOEXEC.BAT, CONFIG.SYS, WIN.INI の書き換えなどである。
 形式は、[first] と同様。ただし、ID は、"Ln" (n = 1, 2, 3, ...) と記述する。

ID	TYPE	COMMAND	ARGS	FILES
L1	D	CHANGE	"AUTO.REG @BOOT@:%autoexec.bat @SI@%autoexec.bat"	

【図11】

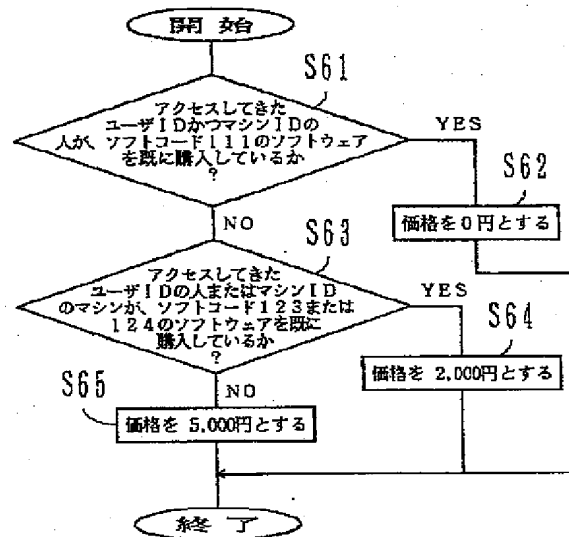
端末上で作成されるファイル群を示す図



【図12】

【図27】

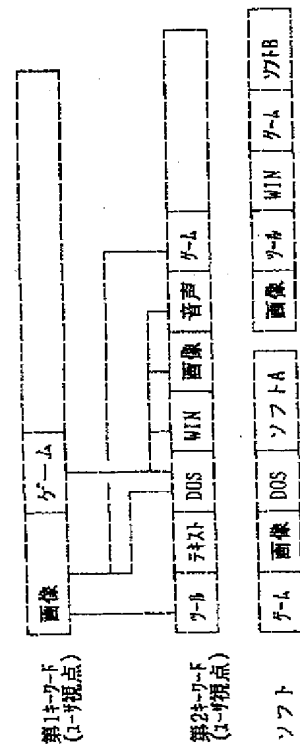
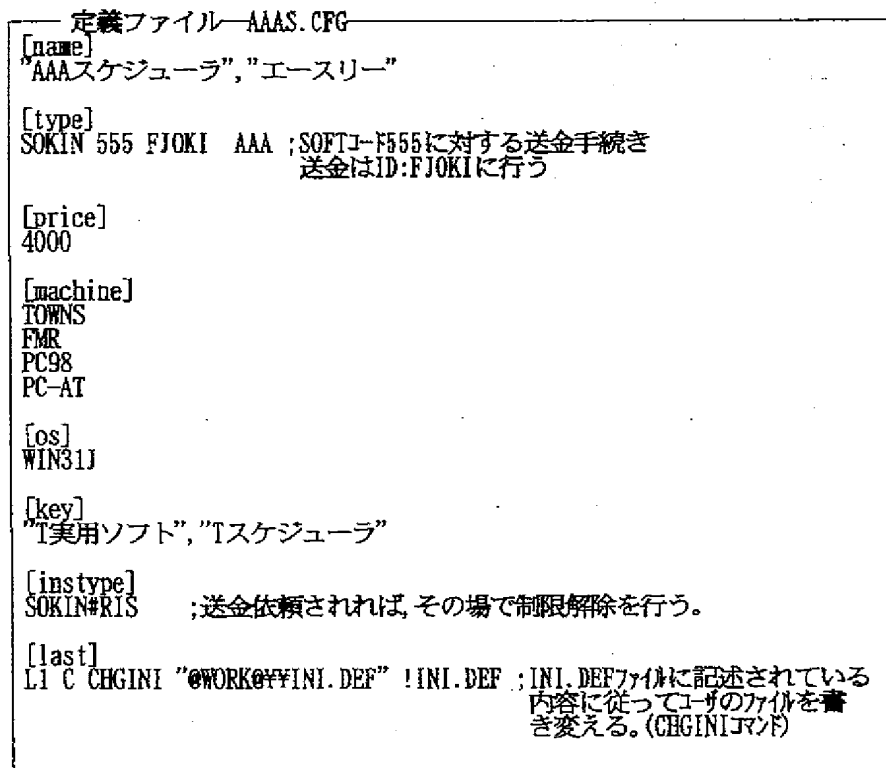
価格決定処理のフローチャート



【図36】

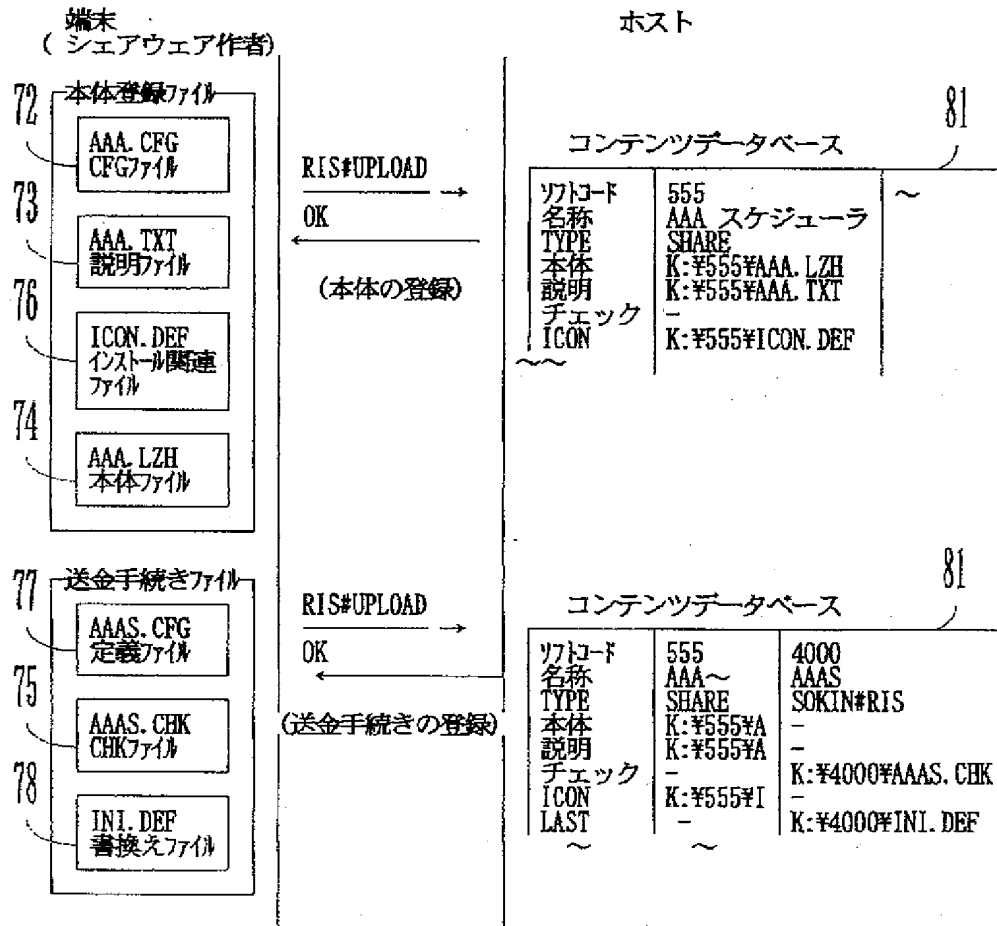
第1の定義ファイルを示す図

キーワードの表示例を示す図



【図15】

アップロードを示す図



【図37】

環境情報の送信プロトコルを示す図

```

RIS_SENDENV
MACHINE:
MODEL=1. (端末ソフトのインストール時に取得)
OS=1.
DRV:
DRVKIND="HS",
HDINF:
HDID="D",
PARTINF:
PARTNO="1",
CAPACITY="50000000",
VACANT="30000000",
OSNAME="MSDOS",
DRVNAME="D",
PRIORITY="1"

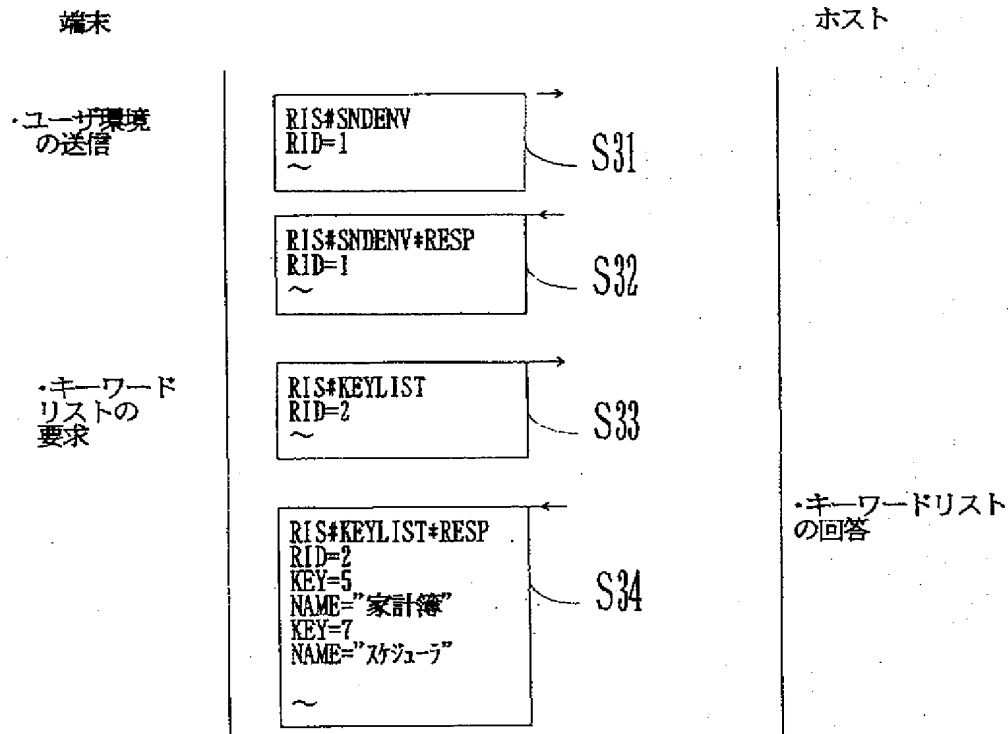
```

(ホストアクセス時に取得)

::

【図16】

シェアウェア手続きを示す図（その1）



【図40】

設定ファイルの例を示す図

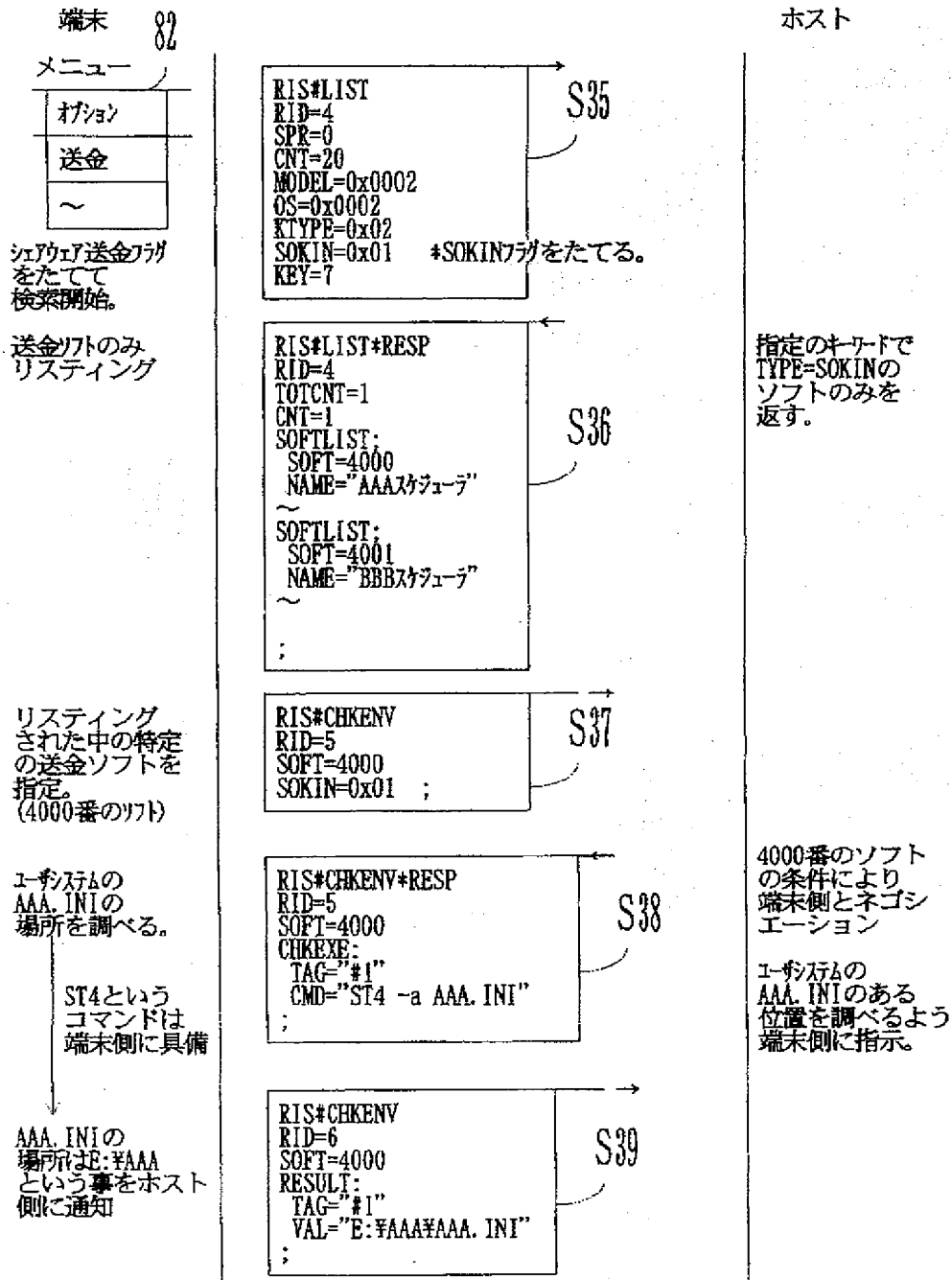
```

[DstDirs]
1=D:\RIS\KOBUTA
2=D:\WINDOWS\SYSTEM
  
```

[Files]	Description,	Dst (Dir:Name),	Shared
Soft2.Exe,		1:Soft2.Exe,	N
TOWNS.DRV		2:FONT.DRV	N

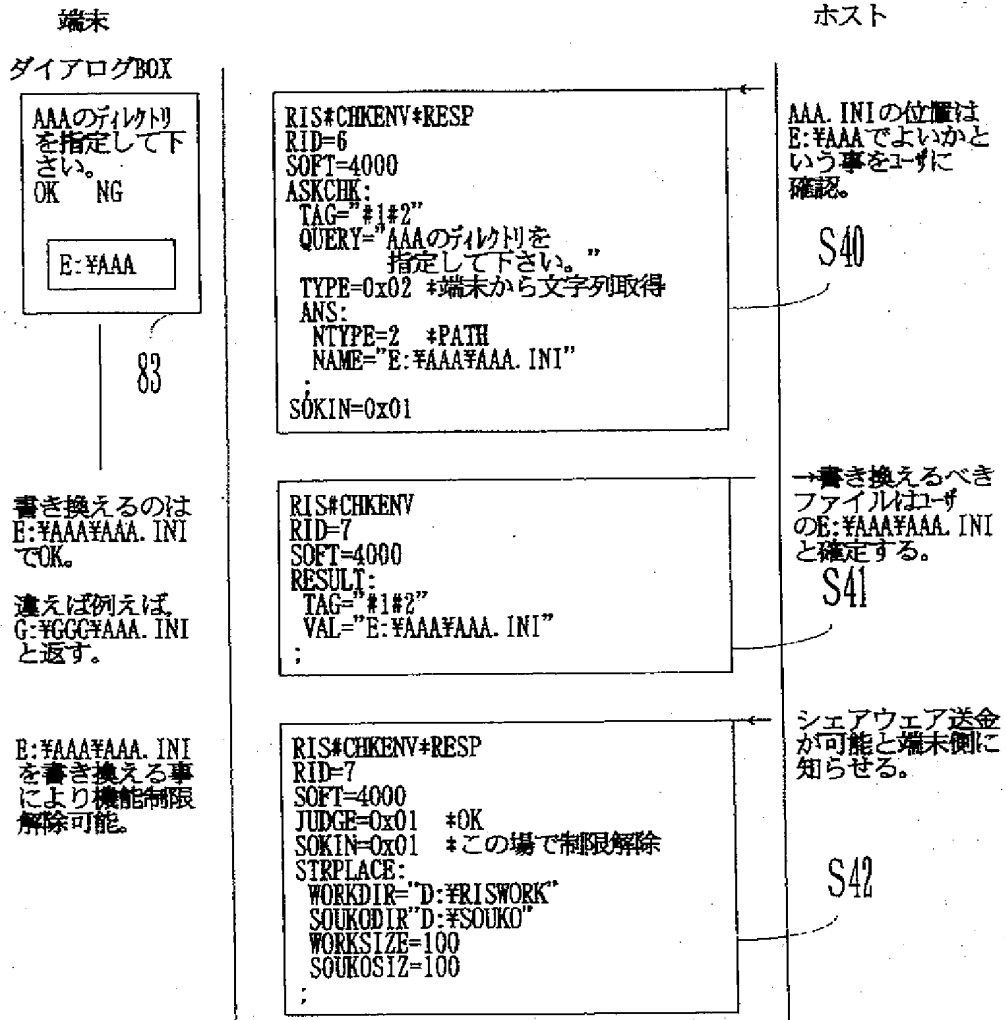
【図17】

シエアウェア手続きを示す図 (その2)



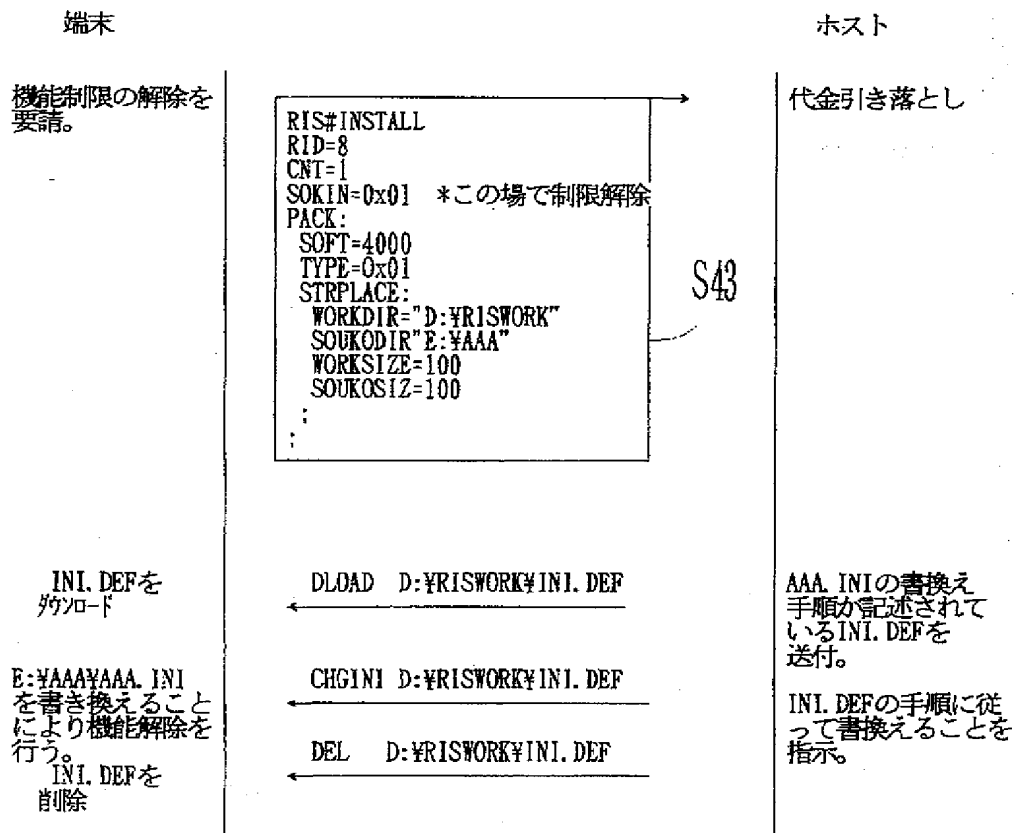
【図18】

シェアウェア手続きを示す図 (その3)



【図19】

シェアウェア手続きを示す図（その4）



【図21】

第2の定義ファイルを示す図

定義ファイル—BBBS.CFG—

```

[name]
"BBBスケジューラ","ビースリー"

[type]
SOKIN 555 FJOKI BBB

[price]
4000

[machine]
TOWNS
FMR
PC98
PC-AT

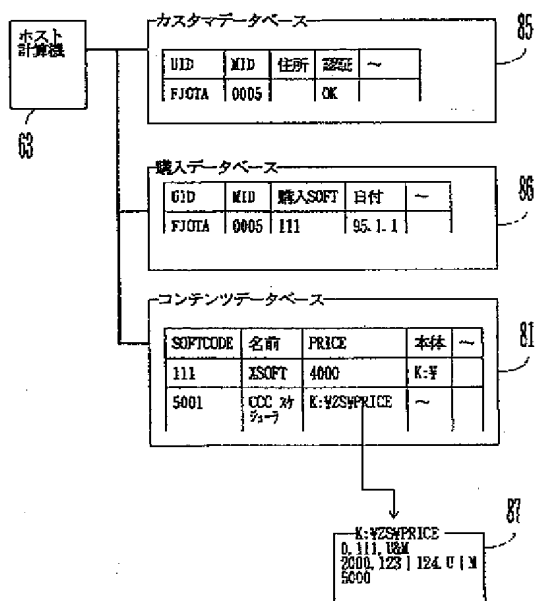
[os]
WIN31J

[key]
"T実用ソフト","Tスケジューラ"

[instype]
SOKIN#MAIL ;送金処理のみでその場での制限解除は行わない
  
```

【図26】

ホスト計算機のデータベースを示す図



【図35】

キーワード選択のプロトコルを示す図

(コマンド) RIS_KEYLIST

(レスポンス) RIS_KEYLIST*RESP
KEYLIST:
KEY=1,
NAME="OS/基本ソフト",
KEY=2,
NAME="開発支援",
KEY=3,
NAME="ゲーム",
～

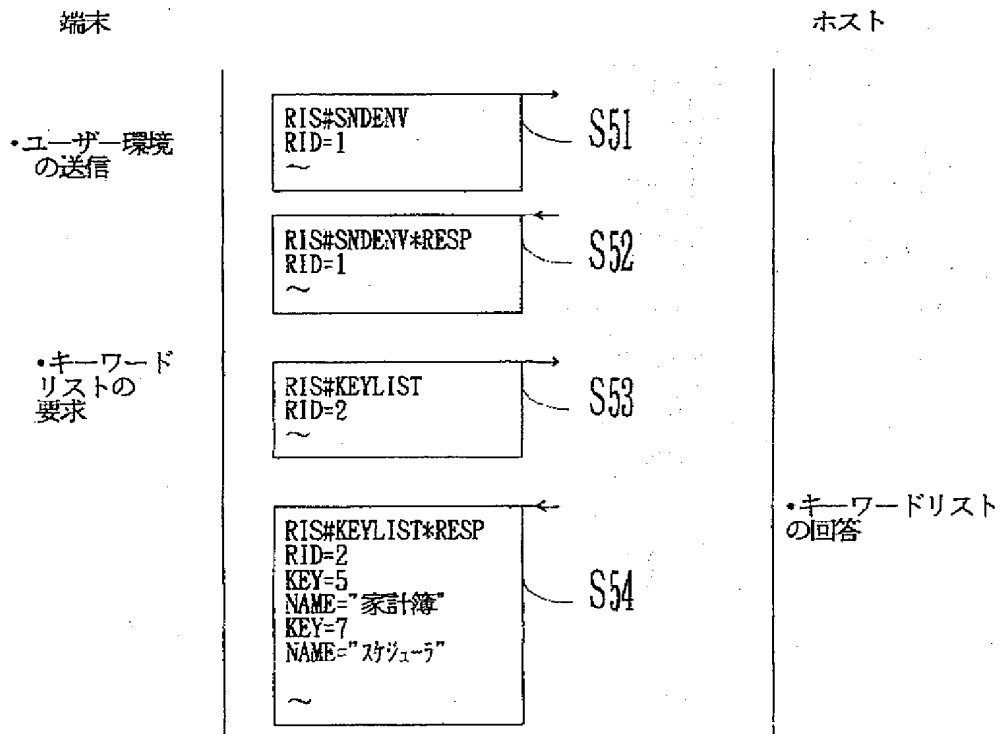
(コマンド) RIS_KEYLIST
KEY=3

(レスポンス) RIS_KEYLIST*RESP
KEYLIST:
KEY=51,
NAME="RPG",
KEY=52,
NAME="アクション",
KEY=53,
NAME="パズル/クイズ",
KEY=54,
NAME="シミュレート",
KEY=55,
NAME="元談",
～

(コマンド) RIS_LIST
KEY=52,
KEY=53

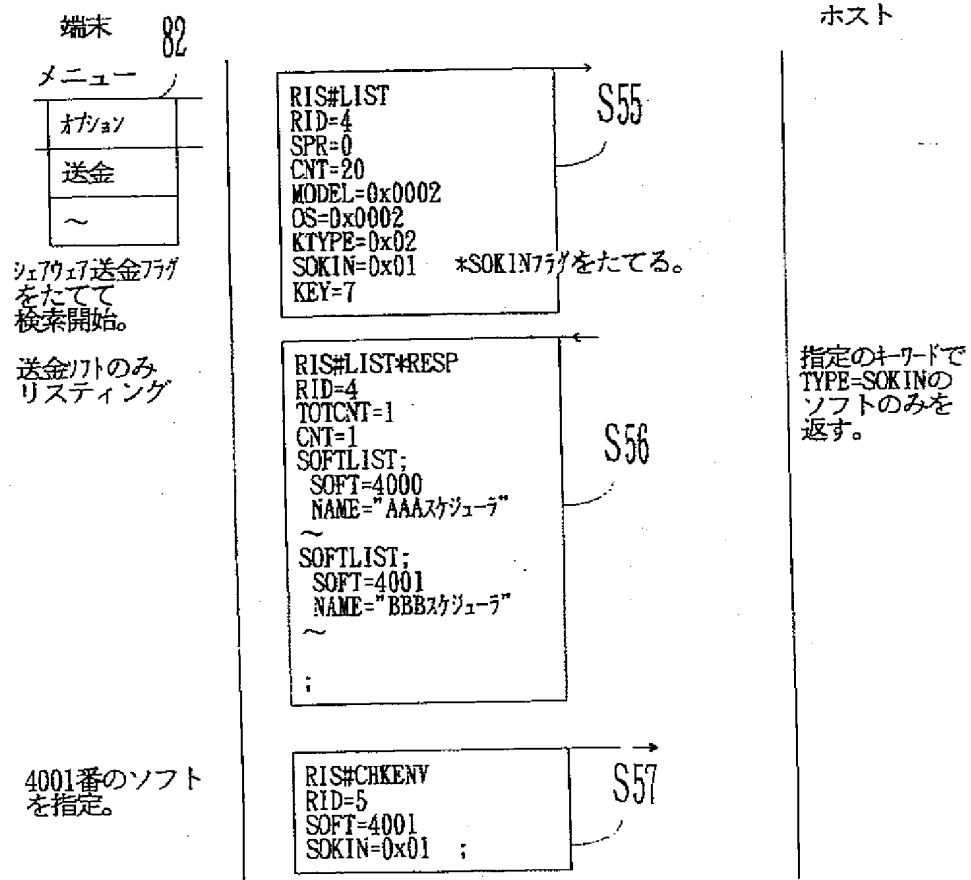
【図22】

代金引き落とし手続きを示す図（その1）



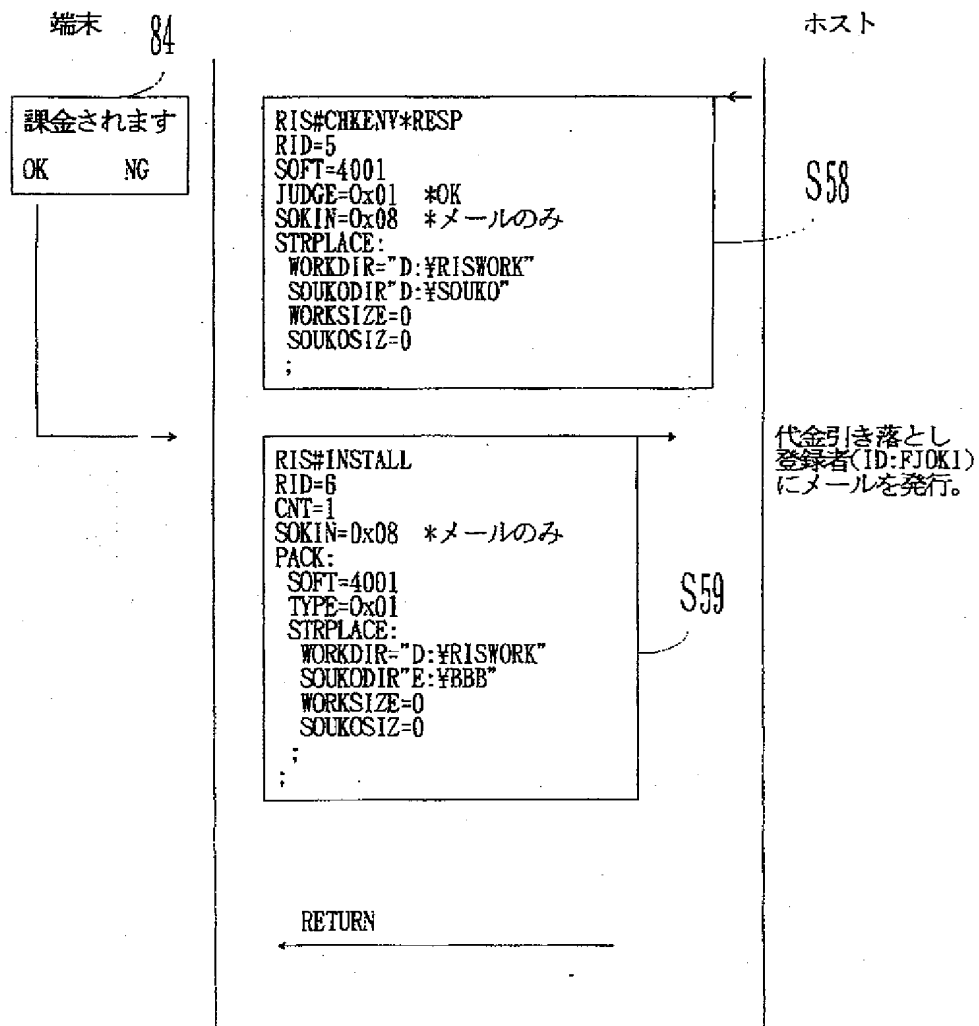
【図23】

代金引き落とし手続きを示す図（その2）



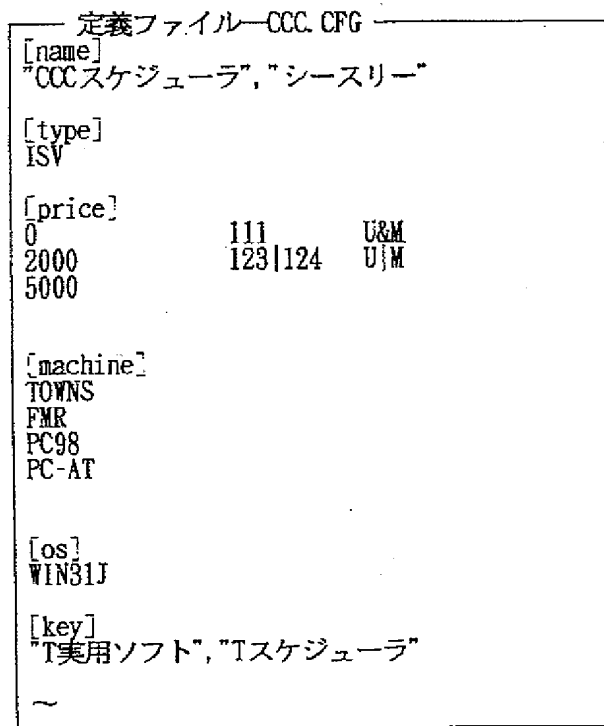
【図24】

代金引き落とし手続きを示す図（その3）



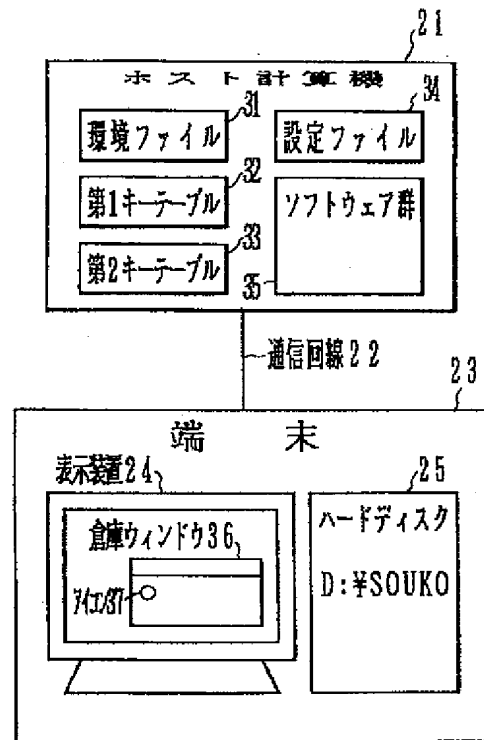
【図25】

第3の定義ファイルを示す図



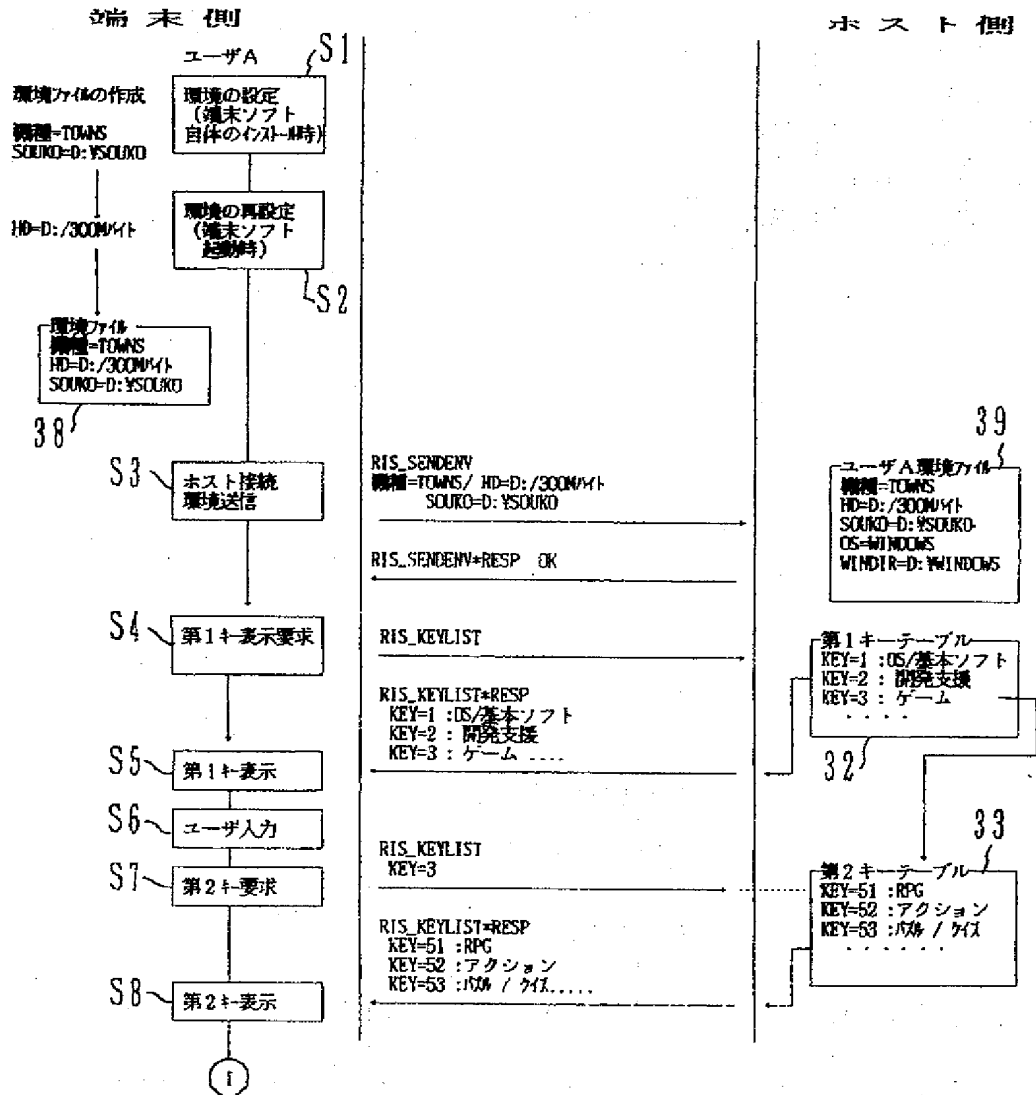
【図28】

先願のリモートインストールシステムの構成図

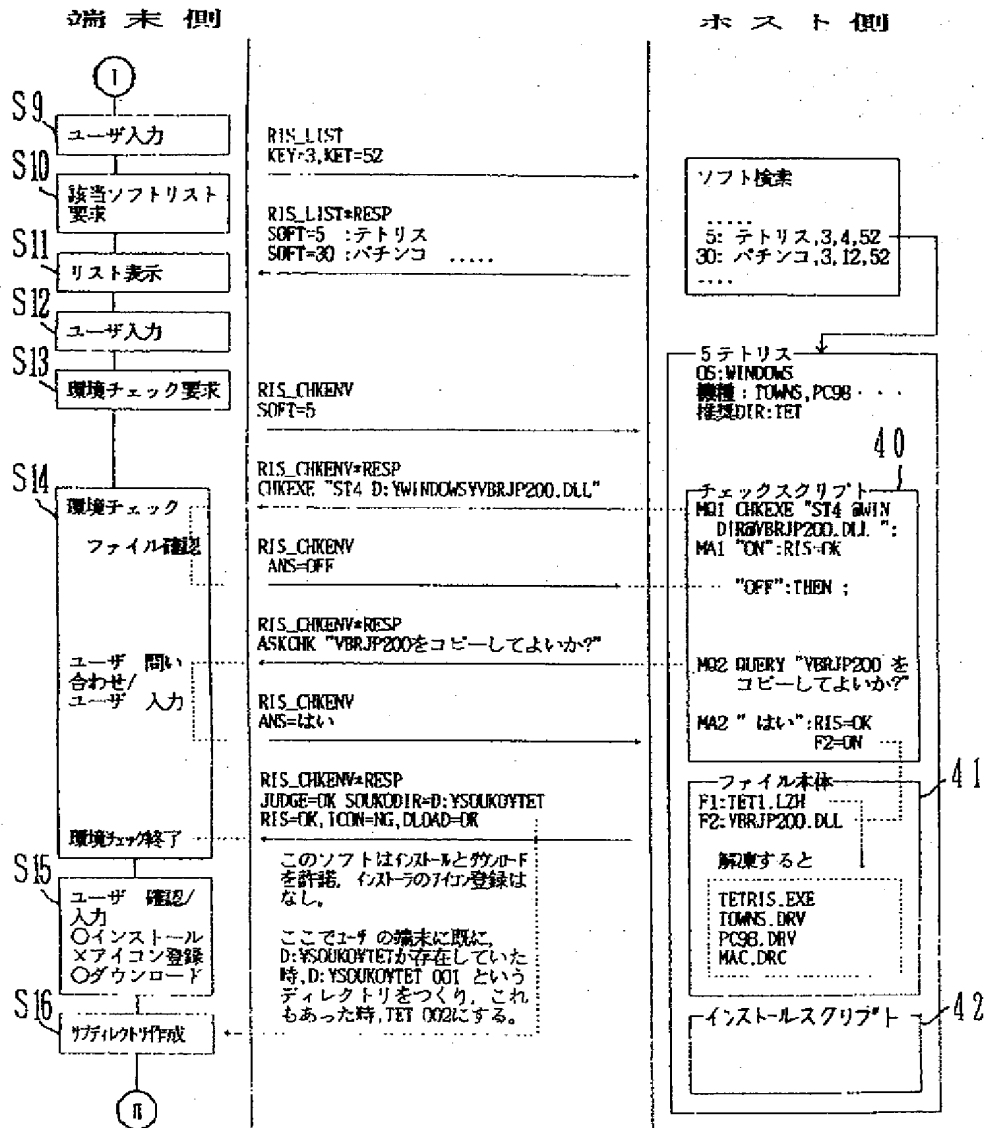


【図29】

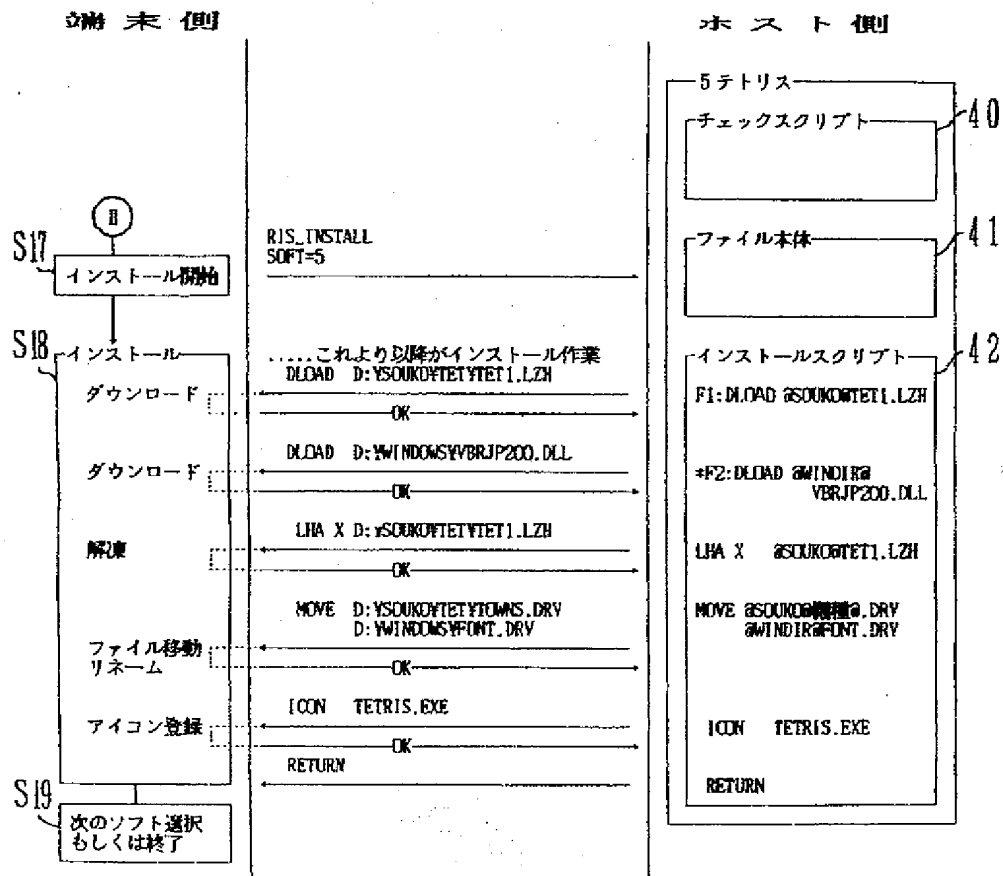
リモートインストールのフローチャート(その1)



リモートインストールのフローチャート(その2)



リモートインストールのフローチャート(その3)



【図32】

【図33】

インストール方法選択ウィンドウの例を示す図 リモートインストールプロトコルを示す図(その1)

システム登録 ○ インストーラのアイコン登録 ○ ダウンロード

```

RIS_SENDENV(端末→ホスト)
RID=1,
MID=1,
TIME=1994/08/94/12/20/20,
MACHINE:
MODEL=1,
OS=1,
DRV:
  DRVKIND=HS,
  HDINF:
    HDID=0,
    PARTINF:
      PARTNO=1,
      CAPACITY=400000,
      VACANT=200000,
      OSNAME=MSDOS2,
      DRVNAME=D,
      PRIORITY=?
    ;
    PARTINF:
      PARTNO=2,
      CAPACITY=500000,
      VACANT=300000,
      OSNAME=MSDOS,
      DRVNAME=E,
      PRIORITY=?
    ;
  ;
  DRVKIND=HS,
  HDINF:
    HDID=1,
    PARTINF:
      PARTNO=1,
      CAPACITY=500000,
      VACANT=200000,
      OSNAME=MSDOS,
      DRVNAME=F,
      PRIORITY=?
    ;
    PARTINF:
      PARTNO=2,
      CAPACITY=700000,
      VACANT=300000,
      OSNAME=MSDOS,
      DRVNAME=G,
      PRIORITY=?
    ;
  ;
  DRVKIND=F2,
  DRVKIND=F2,
  DRVKIND=C
;
SOURKDIR=D:\RISYS\SOUKO, <-----ここで倉庫ディレクトリを指定
WINDIR=D:\WINDOWS,
MEH
WMEH=8000000,
TPA=400000
;
MPSWD="1234"

```

【図38】

環境チェックのプロトコルを示す図

```

RIS_CHKENV
  SOFT=5      (ソフトコード5番をチェックしたい)

RIS_CHKENV*RESP
  CHKEXE:
    TAG="VBRJP200.DLL"      (VBRJP200.DLLというファイルがシステムディレク
    CMD="ST4 D:\WINDOWS\SYSTEM\VBRJP200.DLL"      トリにあるか調査)
  ;
  WORKDIR="D:\RIS\YKOBUTA"
  SOURKODIR="D:\RIS\YKOBUTA"
  ;
  WORKSIZ=700000,
  SOURKSIZ=500000

RIS_CHKENV
  SOFT=5,
  RESULT:
    TAG="VBRJP200.DLL",      (VBRJP200.DLLというファイルはシステムディレクトリにない)
    VAL="OFF"
  ;

RIS_CHKENV*RESP
  ASKCHECK:
    TAG="Q1",      (ユーザに問いあわせる)
    QUERY="
      このソフトを実行するためには、VBRJP200.DLLが必要ですが、
      あなたのシステムにはこのファイルが見つかりません。
      このままでは実行できませんが、インストールしてもよろしいですか？",
    TYPE=0,
    ANS:
      CODE="OK",
      NAME="OK"
    ;
    ANS
      CODE="CANCEL",
      NAME="CANCEL"
    ;

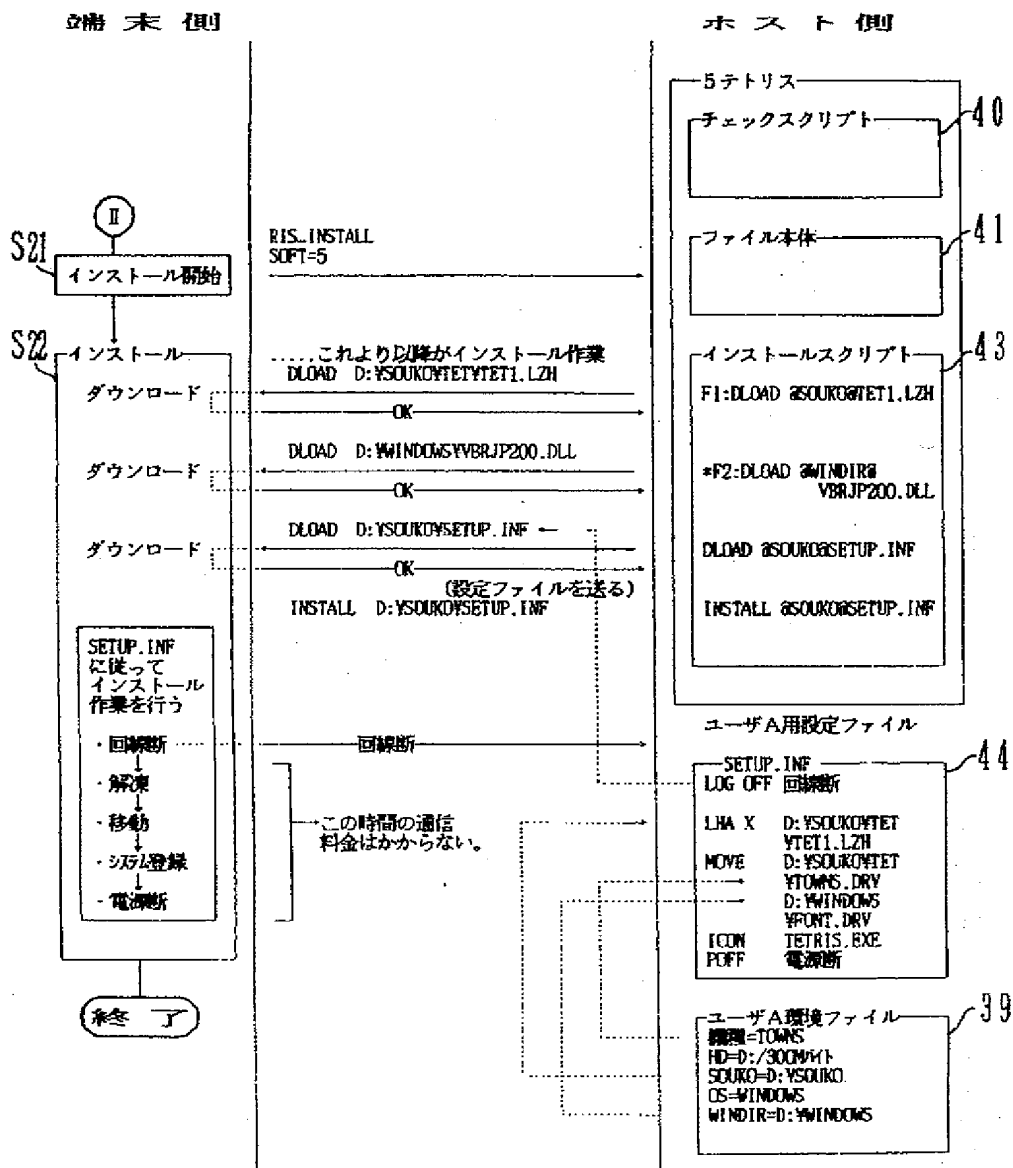
RIS_CHKENV
  SOFT=5,
  RESULT:
    TAG="Q1",      (ユーザがインストールしてもよいと答える)
    VAL="OK"

RIS_CHKENV*RESP
  JUDGE="OK"      (インストール可能)

RIS_INSTALL
  CNT=1,
  TYPE="RIS",
  STRPLACE:
    SOFT=5,
    WORKDIR="D:\RIS\YKOBUTA",
    SOURKODIR="D:\RIS\YKOBUTA"
  ;

```

自動インストールのフローチャート



(72)発明者 岡田 利司郎
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内

(72)発明者 山崎 利哉
神奈川県川崎市中原区上小田中1015番地
富士通株式会社内